게임엔진 이해와 실습

컴퓨터、소프트웨어공학과 20152969 김동관

INDEX

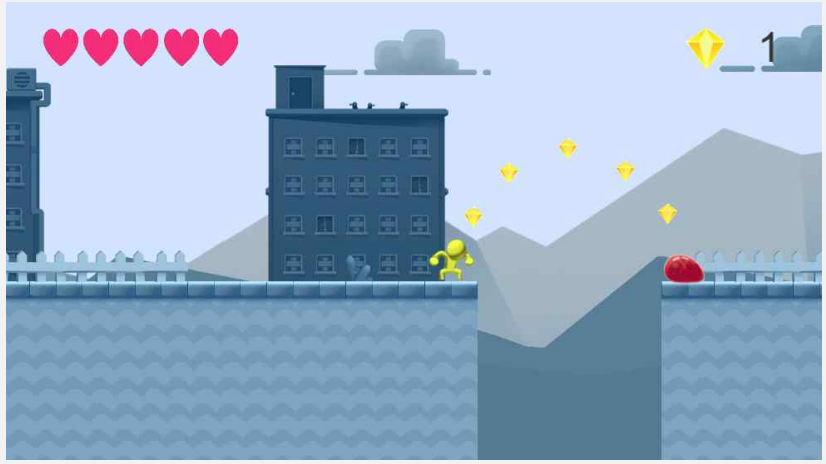
1. 게임 소개
2. 게임의 수정 사항
3. 수정 보완 방법
4. 앞으로 더 보완하고 싶은 점

게임 소개

컴퓨터 • 소프트웨어공학과 20152969 김동관

<시티런>

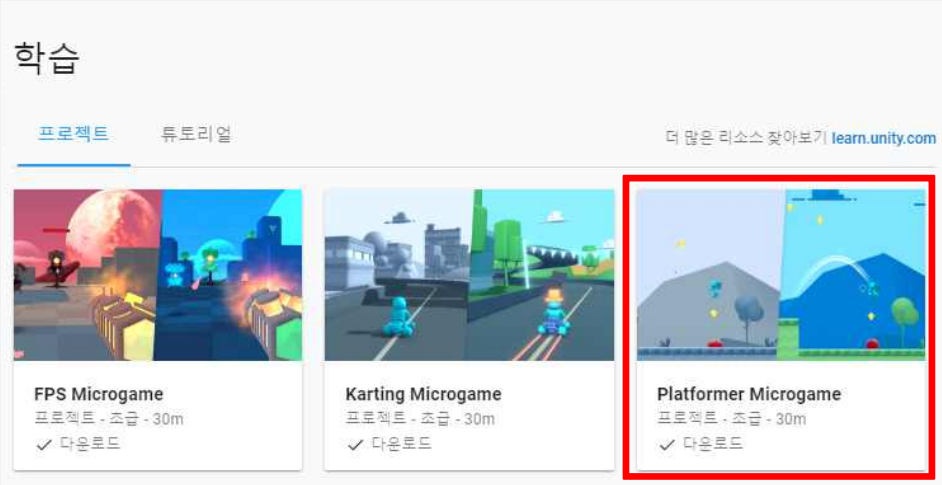
➢ 도심 속 맵을 달리며, 토큰을 모아 안전하게 골인지점까지 달리는 게임.



• 토큰을 모으며 골인지점까지 달리는 점.
• 몬스터의 머리를 밟아 없애는 점.

➢ 슈퍼마리오와 매우 유사하다고 생각했다.

◆ 처음 게임을 받은 곳
[Unity Hub] - [학습] - [Platformer Microgame]



게임의 수정 사항

컴퓨터 • 소프트웨어공학과 20152969 김동관

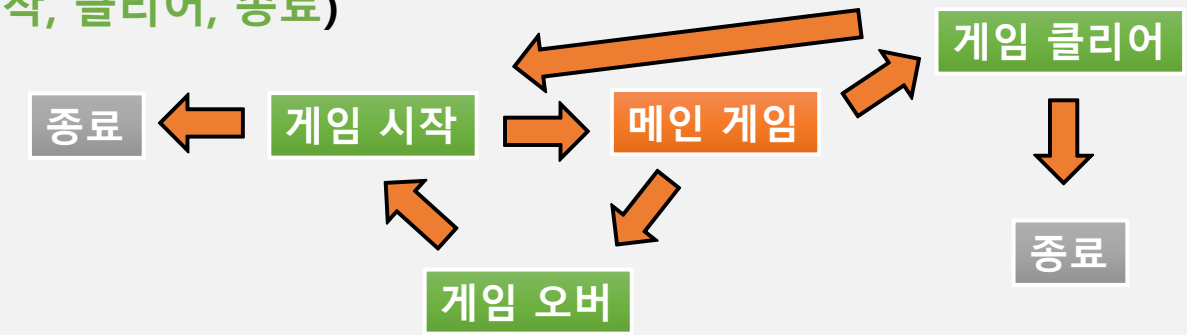
1.캐릭터의 체력 추가

캐릭터의 체력을 5로 설정하여 슬라임에게 5번까지는 부딪혀도 게임이 계속해서 진행됨.
+ 왼쪽 상단 하트 5개 추가 → 체력에 따라 하트 감소 (구멍에 빠지면 바로 게임종료!)

2. 토큰 점수 추가

게임 중 노란색 토큰을 수집하게 되는데 수집한 토큰의 개수를 게임 화면에 표시함.

3. Scene 추가 (게임의 시작, 클리어, 종료)



수정 사항 보완 방법 (체력 추가)

컴퓨터 • 소프트웨어공학과 20152969 김동관

❖ 기존의 게임 : 캐릭터가 구멍에 빠지거나 슬라임에게 부딪히면 첫 시작지점으로 리스폰 된다.
→ 게임의 난이도가 높음.

❖ 게임 수정 : 캐릭터의 체력을 추가하여 5번의 생명을 가짐.

```
public class Health : MonoBehaviour
{

    AudioSource pAe;

    public static int  HP;
    public Image hpb;
```

✓ pAe : 캐릭터와 슬라임이 충돌했을 때 나는 효과음 (오디오 소스)
✓ HP : 캐릭터의 체력
✓ hpb : 하트 이미지 (캐릭터의 체력바)

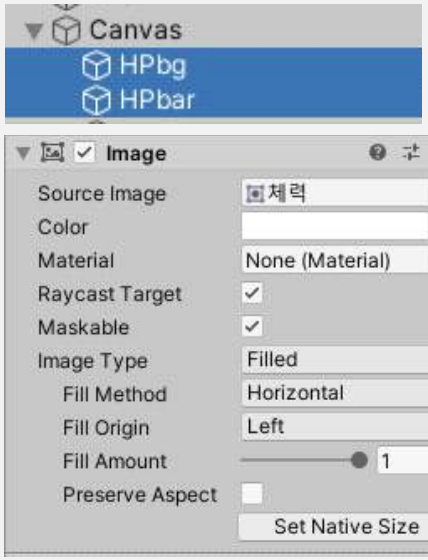
```
void Start()
{
    pAe = GetComponent<AudioSource>();

    HP = 5;
    hpb = GameObject.Find("HPbar").GetComponent<Image>();
}
```

✓ HP를 5로 설정
✓ 오디오 소스와 하트 이미지를 변수에 입력

수정 사항 보완 방법 (체력 추가)

컴퓨터 • 소프트웨어공학과 20152969 김동관

❖ 게임 추가 : 캐릭터의 체력을 나타내는 하트 게이지 추가



✓ HPbg : 하트 배경이미지 (깎인 상태)
✓ HPbar : 하트 배경이미지 (채워진 상태)

<분홍색 하트(HPbar)의 Inspector>
✓Image Type : [Filled]로 설정하여 이미지를 채우는 효과 생성
✓ Fill Origin - [Left] : 왼쪽을 기준으로 게이지가 채워짐
✓ Fill Amount (1) : 1이 게이지의 전체(100%)가 됨.

```
public void MinusHP()
{
    pAe.Play();
    HP--;
    hpb.fillAmount = HP * 0.2f;
```

✓ 충돌로 인한 HP감소 및 효과음 재생
✓ fill.Amount로 게이지 조절
✓ 0.2f : 20%씩 → 하트 1개씩

```
public class PlayerEnemyCollision : Simulation.Event<PlayerEnemyCollision>
{
    public EnemyController enemy;
    public PlayerController player;

    PlatformerModel model = Simulation.GetModel<PlatformerModel>();
```

```
        GameObject.Find("Player").GetComponent<Health>().MinusHP();
```

✓ 캐릭터와 슬라임의 충돌을 관리하는 스크립트
✓ Player(캐릭터)에 연결된 Health 스크립트에서
✓ MinusHP 함수를 호출

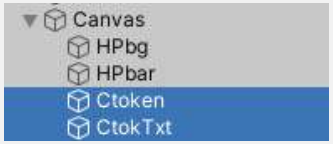
수정 사항 보완 방법 (토큰 점수)

컴퓨터 • 소프트웨어공학과 20152969 김동관

❖ 기존의 게임 : 토큰이 있지만 먹으면 그저 없어질 뿐이다.
→ 게임의 중요한 요소로 작용하지 않음.

❖ 게임 수정 : 수집한 토큰의 수를 오른쪽 상단에 표시하여 점수 역할을 하도록 함.

```
public class TokenController : MonoBehaviour
{
    int tkScore = 0;
    Text tks;
```



✓ tkScore : 수집한 토큰 점수
✓ tks : 수집한 토큰의 점수를 나타내는 텍스트 UI의 변수
✓ Ctoken : 수집한 토큰을 나타내는 이미지
✓ CtokTxt : 수집한 토큰의 점수를 나타내는 텍스트

```
void Start()
{
    tks = GameObject.Find("CtokTxt").GetComponent<Text>();
}
```

```
void Update()
{
    tks.text = tkScore.ToString();
```

```
    tkScore++;
```

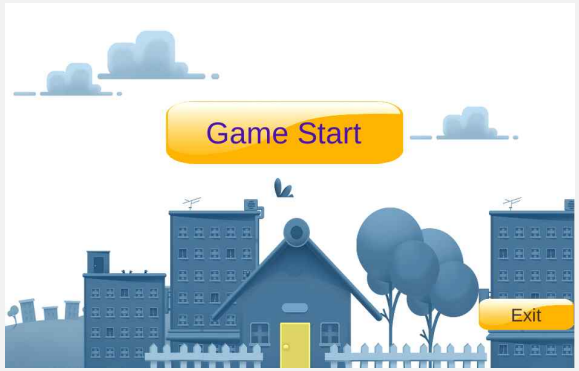
✓ tks 변수에 CtokTxt (텍스트 UI) 입력
✓ 텍스트 UI에 변수 값 대입
✓ ToString으로 변수값을 텍스트로 변환
✓ tkScore++ : 토큰을 수집할 때마다 점수가 오르도록 설정

수정 사항 보완 방법 (Scene 추가)

컴퓨터・소프트웨어공학과 20152969 김동관

❖ 기존의 게임 : 게임이 진행되는 메인 Scene 하나만 존재. ("SampleScene")
→ 게임의 시작과 끝이 불분명함.

❖ 게임 수정 : NewGame, GameOver, VictoryGame Scene들을 새로 만들어 추가하고, 연결함.



<NewGame>

<GameOver>

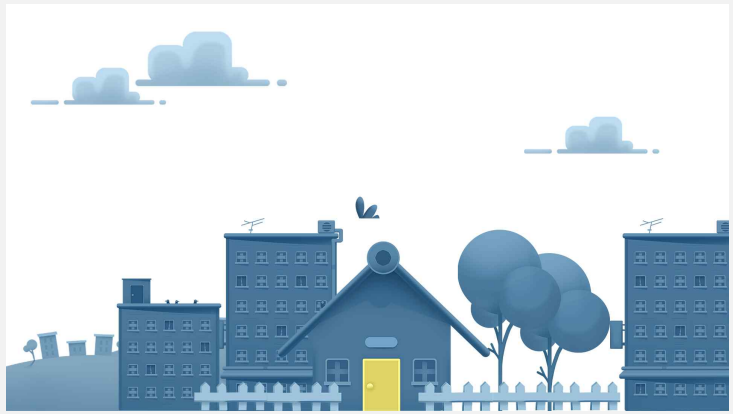
<VictoryGame>

수정 사항 보완 방법 (NewGame)

컴퓨터 • 소프트웨어공학과 20152969 김동관

❖ 게임 추가 : Scene("NewGame") / 배경(Panel) / GameStart 버튼 / Exit 버튼

- ✓ Panel : 게임 배경 이미지
- ✓ StartButton : GameStart 버튼
- ✓ OpTxt : StartButton의 "GameStart" 텍스트
- ✓ ExitButton : Exit 버튼
- ✓ Exit : ExitButton의 텍스트
- ✓ New Game : NewGame 스크립트가 연결된 빈 객체



- ✓ 기존 게임에 있던 이미지를 활용
- ✓ AdobePhotoshop으로 배경이미지와 버튼이미지 제작
- ✓ Panel과 Button에 Image로 배경 삽입

수정 사항 보완 방법 (NewGame)

컴퓨터 • 소프트웨어공학과 20152969 김동관

❖ 게임 추가 : Scene("NewGame") / 배경(Panel) / GameStart 버튼 / Exit 버튼

```
public class NewGame : MonoBehaviour
{
    Text opt;

    public GameObject button;
```

```
void Start()
{
    opt = GameObject.Find("OpTxt").GetComponent<Text>();
    opt.text = "Game Start";
}
```

- ✓ opt : 텍스트(OpTxt) 변수
- ✓ public으로 버튼 선언
- ✓ OpTxt에 스크립트로 텍스트 입력

```
public void Onclick()
{
    SceneManager.LoadScene("SampleScene");
}
```

- ✓ Onclick 함수를 작성
- ✓ GameStart 버튼을 눌렀을 때 이벤트 발생
- ✓ 게임이 실행되는 "SampleScene"으로 전환

```
#if UNITY_WEBPLAYER
    public static string webQuitURL = "http://google.com";
#endif
    public void OnclickExit()
    {
        Quit();
    }
    public static void Quit()
    {
#if UNITY_EDITOR
        UnityEditor.EditorApplication.isPlaying = false;
#elif UNITY_WEBPLAYER
        Application.OpenURL(webQuitURL);
#else
        Application.Quit();
#endif
    }
```

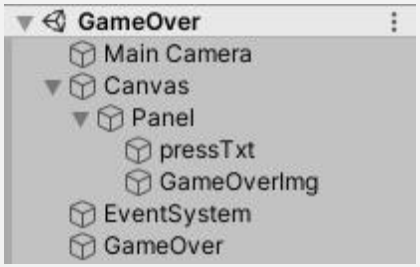
- ✓ webQuitURL : 웹버전의 게임이 종료되었을 때 이동할 URL
- ✓ OnclickExit 함수를 작성
- ✓ Exit 버튼을 눌렀을 때 이벤트 발생 → 게임을 종료함

수정 사항 보완 방법 (GameOver)

컴퓨터・소프트웨어공학과 20152969 김동관

❖ Scene("GameOver") → 캐릭터 HP가 0이 되거나 구멍에 빠지는 경우

❖ 게임 추가 : 아무키를 누르면 NewGame 씬으로 전환



- ✓ Panel : 게임 배경 이미지
- ✓ pressTxt : "Press any key to new game" 텍스트
- ✓ GameOverImg : GameOver 이미지
- ✓ GameOver : GameOver 스크립트가 연결된 빈 객체

❖ Scene 전환 시 문제점 : 캐릭터 HP가 0이 되거나 구멍에 빠지는 순간 바로 전환됨.

→ GameOver Scene으로 전환되기 전 딜레이가 필요함.

```
if(HP == 0)
{
    StartCoroutine(Endgame());
}
```

```
IEnumerator Endgame()
{
    yield return new WaitForSeconds(0.5f);
    SceneManager.LoadScene("GameOver");
}
```

```
void Update()
{
    if (Input.anyKeyDown)
    {
        SceneManager.LoadScene("NewGame");
    }
}
```

```
IEnumerator Ending()
{
    yield return new WaitForSeconds(1.5f);
    SceneManager.LoadScene("GameOver");
}
```

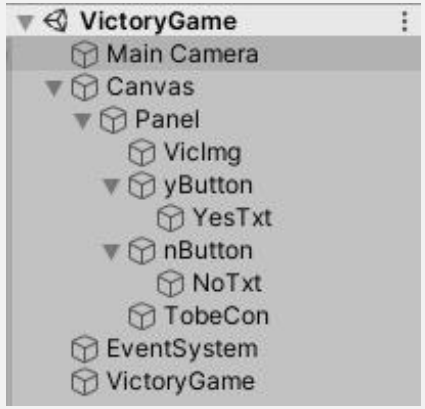
- ✓ Coroutine 사용
- ✓ WaitForSeconds로 딜레이 생성
- ✓ 딜레이 시간 후 Scene 전환
- ✓ HP 0의 경우 0.5초 후 (Health 스크립트)
- ✓ 구멍에 빠진 경우 1.5초 후 (DeathZone 스크립트)

수정 사항 보완 방법 (VictoryGame)

컴퓨터・소프트웨어공학과 20152969 김동관

❖ Scene("VictoryGame") → 게임을 클리어한 경우

❖ 게임 추가 : Victory 이미지 / To be Continue? 텍스트 / Yes 버튼 / No 버튼



✓ Panel : 게임 배경 이미지
✓ VicImg : Victory 이미지
✓ yButton : Yes 버튼
✓ YesTxt : "Yes" 텍스트
✓ nButton : No 버튼
✓ NoTxt : "No" 텍스트
✓ TobeCon : "To be Continue?" 텍스트
✓ VictoryGame : VictoryGame 스크립트가 연결된 빈 객체

```
SceneManager.LoadScene("VictoryGame");
```

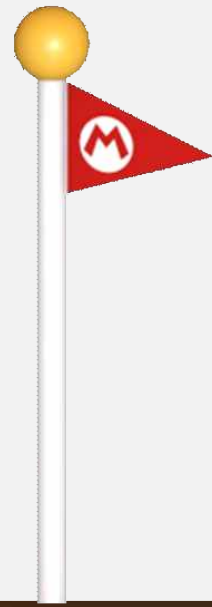
✓ 게임을 클리어한 경우 "VictoryGame" Scene으로 전환
(PlayerEnteredVictoryZone 스크립트)

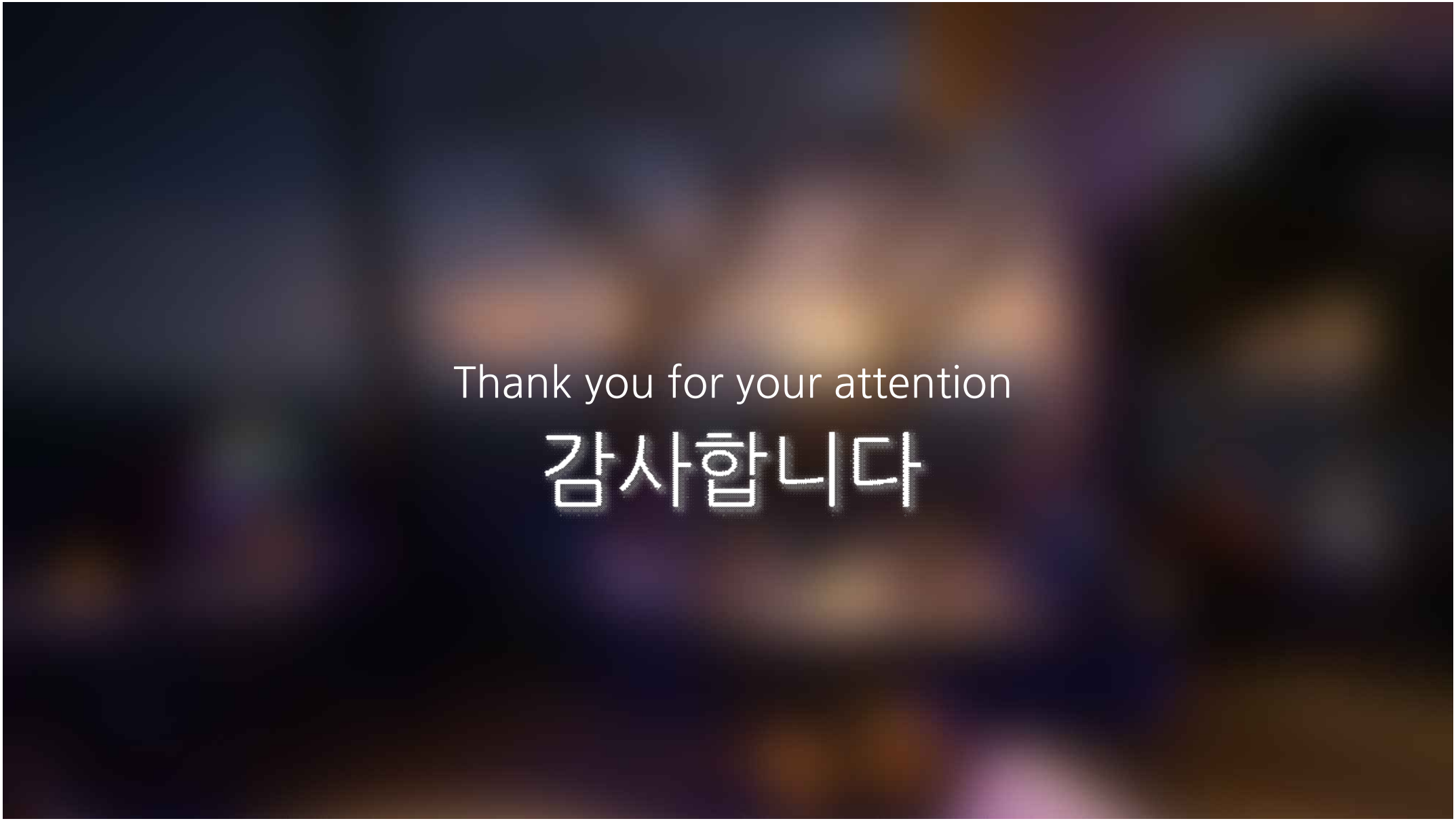
➢ Yes 버튼 → 이벤트 트리거 사용 (NewGame Scene으로 전환)
➢ No 버튼 → 이벤트 트리거 사용 (게임 종료)

※ NewGame Scene의 "GameStart" 버튼, "Exit" 버튼과 동일

앞으로 더 보완하고 싶은 점

- ❖ 캐릭터 추가 → 프리팹 추가
- ❖ 체크포인트 생성 → 기존의 게임의 경우 한 번의 실수로 시작점에서 다시 시작해야함.
- ❖ 이단 점프 구현
- ❖ Stage 추가 → 이단점프와 함께 구현하면 다양한 맵 제작이 가능.
- ❖ 종료화면 랭킹보드 추가 → 토큰의 중요도 상승, 토큰 점수 이용, 슬라임 점수 추가
- ❖ 안드로이드와 iOS 모두 지원




Thank you for your attention
감사합니다