

# With Alone GDD 문서



분류	내용
작성 일자	2024-11-05
작성자	류승재

수정 일자	수정 내역

## 0. Index

---

<b>1. 게임 소개 .....</b>	<b>3</b>
1.1 게임 개요.....	3
1.2 인게임 스크린샷 예시 .....	3
<b>2. 게임 스토리 .....</b>	<b>5</b>
2.1 줄거리.....	5
2.2 주요 캐릭터 개요.....	6
2.3 메인 시나리오 요약 .....	8
<b>3. 게임 메카닉스.....</b>	<b>10</b>
3.1 Scene 표기 규칙.....	10
3.2 게임 플로우.....	11
3.3 게임 시스템 / 콘텐츠.....	12
3.3.1 노벨 게임 파트.....	12
3.3.2 2D RPG 게임 파트 .....	16
3.3.3 멀티 플레이 파트 .....	20
<b>4. 기타.....</b>	<b>21</b>
4.1 With Alone 콘셉트 게임의 한계 .....	21

## 1. 게임 소개

- ▶ 본 게임은 "솔로 플레이"인 줄 알았던 게임이 사실은 "멀티 플레이"였다면? 이라는 게임 시스템을 기반으로 게임 전반을 기획, 제작한 내용입니다.
- ▶ 개인 기획으로 다양한 분야에 도전해 본 기획입니다.

### 1.1 게임 개요

분류	내용
게임 제목	With Alone
게임 장르	비주얼 노벨 장르 + 2D RPG ( 썬꾸르 형식 ) 장르
사용 엔진	유니티
그래픽	2D 도트 그래픽, 탭류
조작키	WASD(이동), E키(상호작용)

### 1.2 인게임 스크린샷 예시



< 2D RPG ( 썬꾸르 파트 ) 예시 이미지 >



< 비주얼 노벨 파트 예시 이미지 >

## 2. 게임 스토리

---

- ▶ 게임의 공간적 배경은 마법이 존재하는 판타지 세계의 국가 "루미나르 제국".
- ▶ 게임의 핵심 갈등 요소는 신분제를 둘러싼 의견 차이.

### 2.1 줄거리

---





신분제도 그리고 신분 해방 운동

- ▶ "루미나르 제국"의 황제는 새로운 개혁으로 일환으로 **신분제**가 아닌 **능력 중심**으로 학생들을 선발하는 "**이펠람 학원**"을 설립한다.
- ▶ 이펠람 학원에서만큼은 평민과 귀족이 모두 평등했으며, 능력에 따라 오히려 평민이 귀족보다 인정받는 경우도 존재했다. 하지만 이러한 새로운 바람은 **귀족과 평민의 갈등**을 유발했고, 학원 내에서도 파벌을 나누어 서로 다툼이 벌어지곤 했다.
- ▶ 그 과정에서 **귀족(왕족)** 중에서 가장 뛰어난 능력을 가진 "**엔젤라**" 둘째 황녀, **평민** 중에서 가장 뛰어난 능력을 가진 "**루시아**"를 둘러싸고 평민과 귀족이 크게 다투는 사건이 벌어지게 되었고, 두 사람은 각자의 입장에서 각자의 방식으로 신분 제도를 둘러싸고 벌어지는 사건에 관여하기 시작한다.

## 2.2 주요 캐릭터 개요

▶ 본 캐릭터 개요는 게임 개발 이후, AI 생성 이미지를 통해 제작되었습니다.

항목	세부 내용
레퍼런스 이미지	
이름	엔젤라
외형적 특징	금발 긴 생머리, 검은 리본 장식
소개	<p>“플레이어 A” 시점의 인물이자 루미나르 제국의 둘째 황녀. 항족 출신으로서 모범을 보일 수 있도록 항상 노력하는 인물. 항상 모범이 될 수 있도록 정갈하고, 평민과 귀족을 가리지 않고 예의 바르게 행동한다. 뛰어난 마법 재능의 소유자로 학원 내에서도 주목받는 인재이다.</p>

항목	세부 내용
레퍼런스 이미지	
이름	루시아
외형적 특징	흑갈색 단발 머리, 흰색 머리핀
소개	<p>“플레이어 B” 시점의 인물이자 이펠람 학원의 몇 안되는 평민 출신. 평민 출신이지만 뛰어난 마법 재능을 인정받아 이펠람 학원에 입학하였다. 이펠람 학원에서 인정받아 평민 출신에서 벗어나 귀족이 되고자 노력하는 인물.</p>

항목	세부 내용
레퍼런스 이미지	
이름	로미(붉은 머리). 루미(푸른 머리)
외형적 특징	[ 로미 ] : 붉은 머리카락, 적극적이고 활발한 성격 [ 루미 ] : 푸른 머리카락, 어른스럽고 사려깊은 성격.
소개	[ 로미 ] 엔젤라 황녀를 보필하는 메이드. 귀족의 입장에서 항상 노력하고, 본받을 점이 많은 엔젤라 황녀를 존경하며, 신분 제도에 대해 어느 정도 찬성하는 입장. [ 루미 ] 루시아를 보필하는 메이드. 뛰어난 재능을 타고났지만 신분제 탓에 재능을 꽃 피우지 못한 루시아 아가씨를 안타깝게 생각하며 신분 제도에 대해 반대하는 입장.

항목	세부 내용
레퍼런스 이미지	
이름	베라 선생님
외형적 특징	안경, 묶어 올린 머리, 카리스마 있는 성격
소개	이펠람 선생님들 가운데에서도 단연 뛰어난 마법 재능을 자랑하는 여 선생. 능력이 있다면 누구든 높은 자리에 오를 수 있다는 이펠람 학원의 모토를 긍정하며 많은 학생들이 자신의 능력을 뽐낼 수 있도록 학생들을 지도하고 있다.

## 2.3 메인 시나리오 요약

---

▶ 본 시나리오는 WithAlone\_황녀루트.pdf 파일을 통해 확인할 수 있습니다.

### [ Prologue ]

게임의 “여신”이 가볍게 게임 속 세계관이 루프하며, 다회차 형식의 게임이라는 것을 암시하는 파트. “선택”의 중요성에 대해서도 간단하게 언급한다.

### [ Day 1 ]

본 세계관에 관한 간략한 설명이 각 플레이어의 메이드를 통해 이루어지고, RPG 파트에서 추가로 메이드의 안내를 통해 자신이 할 수 있는 행동 ( 마법 수련, 공부, 학교 시찰 등 ) 에 관한 기본적인 안내가 주어진다.

### [ Day 2 ]

메이드의 안내를 통해 상대 플레이어 캐릭터에 관한 정보(단, 그 캐릭터는 NPC로 인지 되도록 정보를 제공한다)가 주어지고, 세계관 속 핵심이 되는 소재 “신분제”에 관해서 각 NPC들이 어떤 생각을 가지고 있는지 정보가 제공된다.

이후 평민과 귀족 사이에서 큰 다툼이 벌어지고, 플레이어는 이 다툼에 관해서 “무시한다”, “평민을 돕는다”, “귀족을 돕는다” 세 가지 중 한 가지 선택을 하게 된다.

### [ Day 3 ]

Day 2에 벌어진 사건으로 인해 귀족과 평민이 파벌로 나뉘어 가벼운 신경전을 벌이는 상태가 된다.

각 플레이어의 메이드는 플레이어가 Day 2에 선택했던 선택지에 관한 간단한 코멘트를 남기고, 추후 행동에 관한 안내를 제공한다. 이후 학교 일정이 끝나고, 사건의 당사자들이 각각의 플레이어를 찾아와 귀족 혹은 평민 측을 도와달라는 요청을 하게 된다.

마찬가지로 플레이어는 그 요청을 들어줄지, 들어주지 않을지 선택할 수 있다.

### [ Day 4 ]

평민과 귀족 사이의 갈등은 결국 극한으로 치닫게 되고, 각 플레이어의 능력치에 따라서 결말이 나뉘게 된다.

평민을 지원하는 플레이어의 능력치가 더 높다면 평민 반란이 성공해 반대 플레이어가 감옥에 가는 엔딩이, 귀족을 지원하는 플레이어의 능력치가 더 높다면 귀족의 평민 억압



이 성공해 반대 플레이어가 감옥에 가는 엔딩이, 각 플레이어가 동일한 선택을 했다고 했다면 플레이어 둘 중 한 사람이 본보기로 감옥에 갇히는 엔딩을 맞이하게 된다.

#### [ 채팅 파트 ]

결과적으로 감옥 내에서 서로를 마주보며 마지막으로 솔로 플레이로 진행되는 게임이 사실은 2인 플레이였으며, 각 플레이어의 선택이 엔딩에 영향을 주고 있었다는 사실을 깨닫게 되며, 서로 느낀 감정을 채팅으로 이야기하며 끝나게 된다.

#### [ 히든 엔딩 ]

2D RPG 내에서 특정 기믹을 수행하여 서로의 존재를 알게 된다면 중간에 루트가 바뀌어 서로가 행복한 엔딩을 볼 수 있도록 설계. ( 시간 문제로 인해 실제로 구현되지는 못했음. )

### 3. 게임 메카닉스

---

▶ 게임의 메카닉스에 관한 부분을 기록합니다.

#### 3.1 Scene 표기 규칙

---

H or P	Day 1	1
(1)	(2)	(3)

◇ [ Scene HDay1\_1 ] ( 노벨 게임 파트 )

[ Effect : Fade In ]

[ BCG : 침실 ]

[ SCG : 메이드1 ]

??? : “눈을 뜨셨나요? 엔젤라 황녀님? 좋은 아침이에요.”

##### [1] H or P

시나리오를 진행하고 있는 플레이어의 신분을 나타내는 부분. “H”는 황녀 신분의 “플레이어 A”를, “P”는 평민 신분의 플레이어 B의 플레이어를 나타낸다.

##### [2] Day 1

진행 중인 시나리오의 현재 날짜를 표시하는 부분, “Day 1”부터 “Day 4”까지 존재합니다.

##### [3] 1

같은 날짜 안에서 구분되는 파트를 표시하는 부분. “1”부터 “3”까지 파트가 존재합니다.

[ 파트 1 ] : “노벨 게임 파트”로 등교 전 메이드와 대화를 하는 파트입니다.

[ 파트 2 ] : “노벨 게임 파트”로 등교한 이후 학교에서 벌어지는 일들을 다루는 파트입니다.

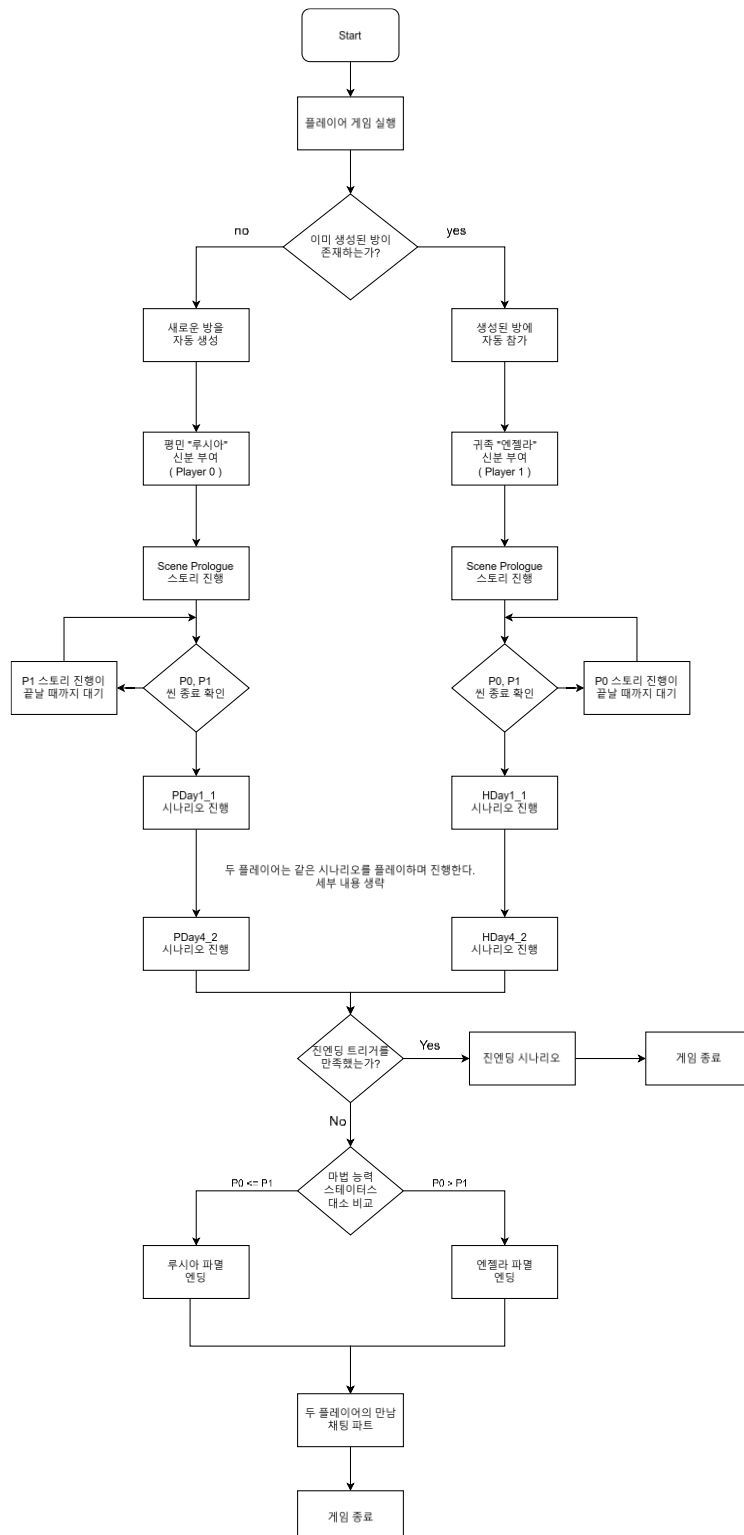
[ 파트 3 ] : “플레이어 조작 파트”로 학교 일정이 모두 끝난 이후 플레이어가 자유롭게 탐사할 수 있는 파트입니다.

▶ 최종적으로 Scene과 파트를 표기할 때는 “Hday3\_2(황녀 Day3 - 2파트)”처럼 표시합니다.

## 3.2 게임 플로우

▶ 게임 진행의 기본적인 흐름을 담은 플로우 차트입니다. 세부 내용은 노션 페이지를 통해 확인해 주세요.

( <https://www.notion.so/With-Alone-88e3a4889fcc47c4a3d2665fb72694be#e5735831497f47d1a6408ed23b83031e> )



### 3.3 게임 시스템 / 콘텐츠

---

- ▶ 게임 내부에 담겨 있는 시스템 / 콘텐츠를 기록한 파트입니다.

#### 3.3.1 노벨 게임 파트

---



- ▶ 비주얼 노벨 시스템에 연관된 다양한 시스템을 구현한 파트입니다.
- ▶ 크게 6가지의 시스템이 구현되어 있습니다.
- ▶ 가장 기본적인 노벨 게임 시스템이 구현되어 있습니다.

## [1] BCG 변경 ( 배경 CG 변경 )

```
void BCGChange(string value)
{
    BCG.sprite = resources.BCGs[int.Parse(value)];
}
```

index	Field	Type	Detail
1	Index	Int	BCG 인덱스
2	FileName	String	BCG 파일 이름

( BCG 기본 데이터 스키마 )

□ Index 값을 받아 특정 BCG를 불러와 배경 CG의 Sprite를 변경해주는 함수입니다.

## [2] SCG 변경 ( 스탠딩 CG 변경 )

```
void SCGChange(string value)
{
    SCG.sprite = resources.SCGs[int.Parse(value)];
}
```

index	Field	Type	Detail
1	Index	Int	SCG 인덱스
2	FileName	String	SCG 파일 이름

( SCG 기본 데이터 스키마 )

▶ 기본적으로는 메인으로 등장하고 있는 캐릭터의 표정을 바꾸는 경우 사용하는 함수.



픽크루 [https://picrew.me/en/image\\_maker/4796](https://picrew.me/en/image_maker/4796) ( 상용 가능 ) 사용 / 순서대로 기본, 약간 화남, 떨어름, 웃음 표정

타 비주얼 노벨 시스템의 경우에는 캐릭터의 위치나 이동 애니메이션 등등 다양한 시스템이 추가되지만, 본 프로젝트에서는 표정을 변경하는 SCG 변경 기능만 구현했습니다.

### [3] 이름 변경 ( 스탠딩 CG 변경 )

```
void NameChange(string name)
{
    if(name == "Nar")
    {
        Name.SetActive(false);
    }
    else
    {
        Name.SetActive(true);
        Name.GetComponentInChildren<Text>().text = name;
    }
}
```

▶ 현재 대화하고 있는 인물의 이름을 표시해주는 시스템. 플레이어의 대사는 "Nar(나레이션)"로 표기해서 이름을 표기하지 않습니다. 그 외는 대사 발원자의 이름 출력합니다.

### [4] 대사 출력 시스템

```
public bool effectOngoing = false;

public IEnumerator TypeEffect(string line, Text text)
{
    effectOngoing = true;
    for(int i = 0; i <= line.Length; i++)
    {
        if (effectOngoing)
        {
            text.text = line.Substring(0, i);
            yield return new WaitForSeconds(0.03f);
        }
        else
        {
            text.text = line;
            yield break;
        }
    }
    effectOngoing = false;
}
```

▶ 대사 출력 시스템은 "타이핑 효과 구현"을 중심으로 구현되었습니다.

가볍게 시스템 플로우를 이야기하자면 이렇습니다.

#### [ 대사 출력 시스템 플로우 ]

[1] 플레이어의 '엔터키'를 입력 받습니다.

[2] 대사가 타이핑 효과를 통해 출력 중인지 확인합니다. ( effectOngoing 변수 )

[2-1] 대사가 출력 중이라면, 코루틴을 중단하고 대사 전체를 한 번에 출력한 이후 effectOngoing을 false로 변경해 타이핑 효과가 끝났음을 알립니다.

[2-2] 대사가 출력 중이 아니라면, 다음 대사(string 변수)를 전달 받아 타이핑 효과를 통해 출력합니다. EffectOngoing 변수를 true로 바꿔 효과가 진행 중임을 알립니다.

[2-2-1] 타이핑 이펙트가 진행 중인 경우 "타이핑 사운드"를 출력합니다.

## [5] 대사 종료 알림 아이콘

- ▶ 출력되고 있는 대사가 모두 출력되었음을 알리는 아이콘입니다.
- ▶ 타이핑 효과가 종료되었을 때, SetActive를 통해 활성화합니다.

## [6] 선택지 시스템

- ▶ 게임 속 플레이어의 선택을 받을 수 있는 버튼을 생성하는 시스템입니다.
- ▶ 이미지에는 5번까지만 번호를 표시했지만, 선택지 시스템도 중요한 시스템이기에 기록했습니다.

### [ 선택지 시스템 플로우 ]

[1] 선택지 시스템이 실행되면, 우선 대화 시스템과 상호작용하는 것을 막아 선택지 시스템 진행 도중 대화를 넘기는 행위를 막습니다.

[2] **선택지의 string 배열** 넘겨 받아, 각 버튼의 Text와 연동해줍니다.

[3] string 배열의 길이를 받아서 선택지의 개수를 넘겨주고, 그만큼 버튼을 활성화합니다.

[4] 플레이어가 버튼을 클릭하면, 그 버튼의 인덱스를 넘겨 받아 스토리 분기점을 변경합니다. 이후 선택지를 전부 비활성화하고, 대화 시스템과 상호작용할 수 있도록 합니다.

[5] 만약 선택 내용에 따라 트리거를 변경해야 한다면, 변동된 스토리 분기점의 첫 대사 에서 트리거 변수를 변경해줍니다.

### 3.3.2 2D RPG 게임 파트

- ▶ Scene 규칙의 [ 3번 파트 ]를 담당하고 있는 2D RPG 게임 파트입니다.
- ▶ 진엔딩의 조건인 타 플레이어의 존재를 알 수 있도록 하는 단서 획득과 엔딩에 영향을 줄 수 있는 스테이터스를 변경시킬 수 있는 파트입니다.



#### [ 플레이어 조작키 ]

조작	Key 할당 혹은 조작 방법
이동키	W A S D
상호작용	E
대사 넘기기	Enter

#### [1] 플레이어의 이동 시스템

- ▶ 플레이어는 W A S D 키를 사용해 상, 하, 좌, 우 방향으로 이동할 수 있습니다.
- ▶ 플레이어는 대각선 방향으로 이동할 수 없습니다.
- ▶ 타일 기반의 그리드 이동은 아닙니다.



## [2] 플레이어의 상호작용

- ▶ 플레이어는 E 키를 사용해 맵에 존재하는 오브젝트와 상호작용할 수 있습니다.
- ▶ Raycast 를 활용해 플레이어가 바라보는 전방 1칸에 있는 오브젝트를 감지합니다.
- ▶ 각 오브젝트는 Object 레이어에 소속되어 있으며, GetMask를 통해 Object를 판별합니다.
- ▶ 상호 작용 가능 오브젝트는 일반적으로 다음과 같은 정보를 가집니다.

index	Field	Type	Detail
1	Index	Int	오브젝트의 고유 번호
2	Name	String	오브젝트의 이름
3	InkJson	TextAsset	각 오브젝트가 가지고 있는 대화 정보가 담긴 파일

### [2-1] 상호작용 가능 콘텐츠

#### [1] NPC와의 대화를 통해 새로운 이벤트의 트리거를 얻기



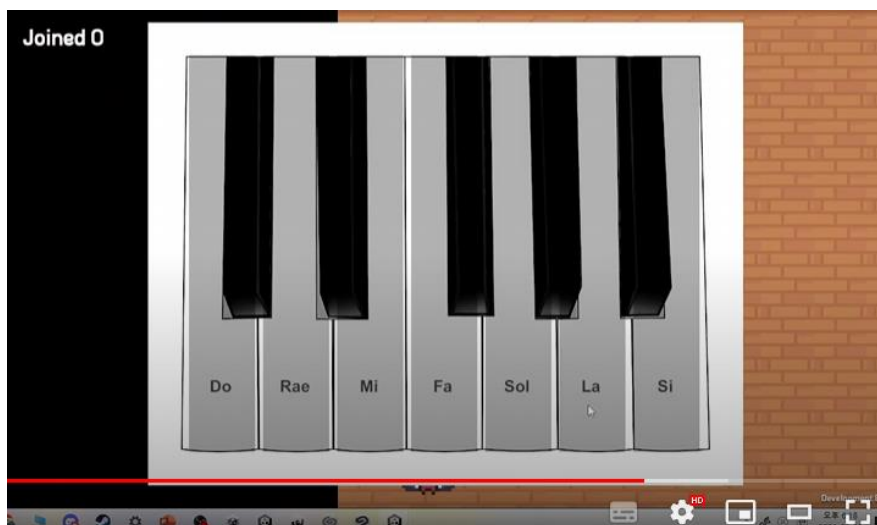
- ▶ 플레이어는 NPC와의 대화를 통해 새로운 스토리를 확인하거나, 새로운 이벤트의 트리거를 얻을 수 있습니다.

## [2] 자신 이외의 다른 플레이어가 존재한다는 단서를 획득

▶ 플레이어는 특정한 오브젝트와의 상호작용을 통해 자신 이외의 다른 플레이어가 존재한다는 단서를 얻을 수 있습니다. 서로가 모든 단서를 획득하면, 진엔딩을 확인할 수 있는 트리거를 얻을 수 있습니다.

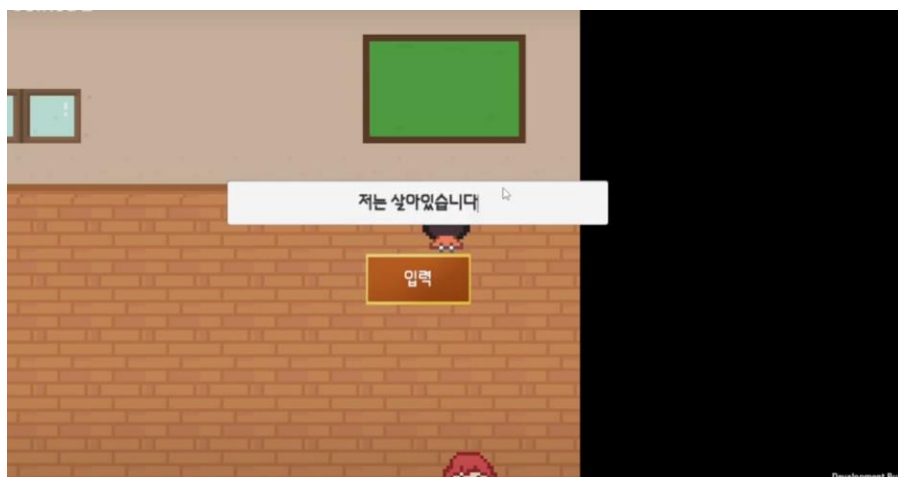
### ① [ 피아노 연주 ]:

맵 상에 존재하는 피아노를 연주하면 자신이 플레이한 피아노의 연주가 상대 플레이어에게 들리게 됩니다. 이를 통해 자신 이외의 다른 플레이어가 존재한다는 것을 눈치챌 수 있도록 설계했습니다.



### ② [ 방명록 확인 ]

학생회의 의견을 모집하는 방명록을 통해서 플레이어는 자신의 메시지를 남길 수 있습니다. 만약 상대 플레이어가 메시지를 남겼다면 플레이어가 방명록을 확인했을 때 남긴 메시지를 확인할 수 있습니다.



[3] 플레이어의 스테이터스에 연관된 스토리를 진행하고, 파트 3를 종료하고 다음 진행 파트로 넘어가기



- ▶ 엔딩에 영향을 주는 플레이어의 능력 스테이터스에 연관된 스토리 진행
- ▶ 만약 이 행동을 선택한다면, 오늘 중 다른 일정은 선택할 수 없다는 안내를 통해 파트 3가 종료된다는 것을 안내합니다.
- ▶ 원 기획에서는 다양한 스테이터스 조합에 따라 다양한 결말을 볼 수 있게 하려고 했지만, 최종적으로는 단순히 마법 능력치의 단순 대소 비교를 통해 지정된 결말을 볼 수 있도록 만들었습니다.

### 3.3.3 멀티 플레이 파트

▶ 멀티 플레이 시스템은 크게 두 가지, "방 생성 파트"와 "채팅 파트"로 나뉩니다.

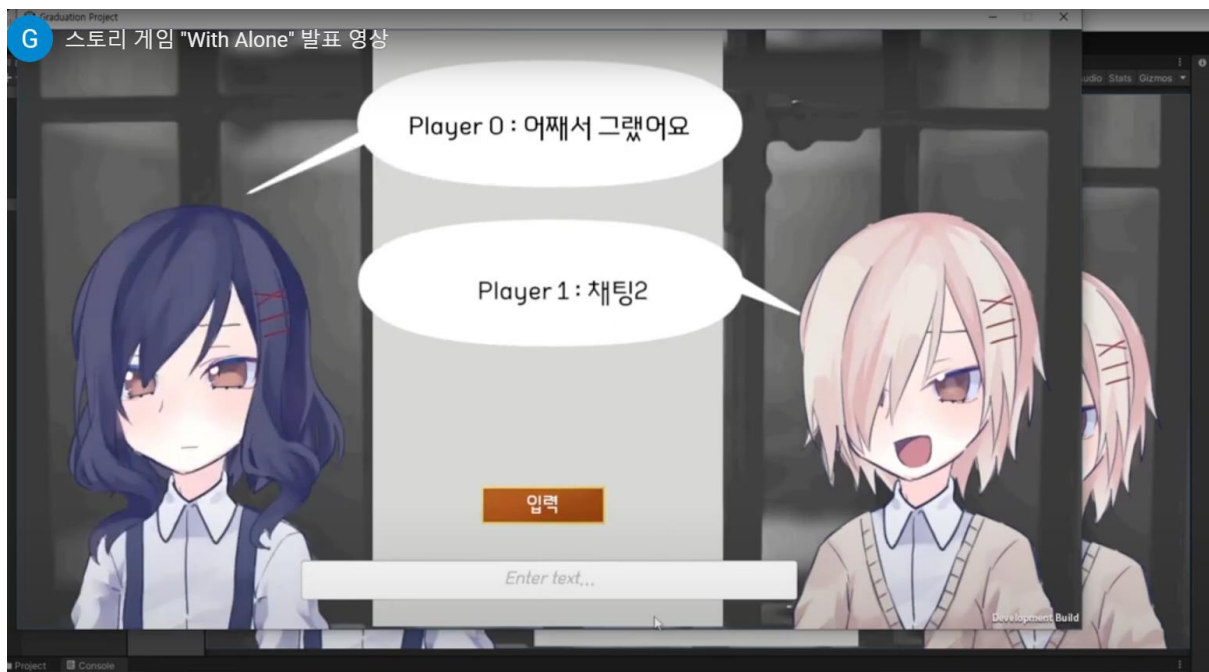
#### ① 방 생성 파트

```
public override void OnJoinRandomFailed(short returnCode, string message)
{
    Debug.Log("활성화된 방이 없으므로 방을 생성합니다.");
    Debug.Log("방장 플레이어가 왕녀 신분으로 게임에 참가하게 됩니다.");
    playerData.player_rank = 0;
    player_rank = 0;
    PhotonNetwork.CreateRoom("Random Room", new RoomOptions { MaxPlayers = 2 });
}
```

▶ 참가할 수 있는 방이 없다면 플레이어는 "Rank 0"를 부여받고, 총 인원이 2명 제한으로 생성된 방을 생성한 이후 평민 신분의 루시아로서 게임에 참가하게 됩니다.

▶ 서로 게임을 시작할 수 있는 상태가 되었다면(프롤로그 대화 종료), 시스템이 서로의 변수를 체크해 다음 진행으로 두 플레이어를 이동시킵니다.

#### ② 채팅 파트



▶ 각 플레이어가 사실은 이 게임이 "멀티 플레이"였다는 것을 알아채는 첫 파트입니다.

▶ 채팅은 누적되지 않고, 입력하는 즉시 새로운 채팅을 갱신합니다.

## 4. 기타

---

- ▶ 그 외의 부분을 다룹니다.

### 4.1 With Alone 콘셉트 게임의 한계

---

- ▶ 본 게임은 개인적으로는 게임 그 자체만으로는 약점이 많은 게임이라고 생각합니다.
- ▶ 게임을 진행하면서 생길 수 있는 약점들은 다음과 같습니다.
  - ➔ 플레이어가 중간 게임을 빠져나왔을 경우의 처리
  - ➔ 플레이어의 글을 읽는 속도가 각각 달라 게임 진행 상황이 일치하지 않는 경우의 처리
  - ➔ 채팅 시스템 진행 도중 플레이어의 비속어나 언어 폭행에 관한 처리
  - ➔ 진엔딩을 보기 위해 반복 플레이를 유도한다는 점
  - ➔ 기타 등등
- ▶ 기본적으로 다양한 예외 처리에 대응할 수 없는 게임이기 때문에 단발적인 이벤트 현장이나 공익을 목적으로 하는 오프라인 캠페인 현장에서 “당신이 무심코 저지른 선택이 사실은 상대방에게 큰 영향을 줄 수 있다”는 교훈을 주는 게임으로 제작하는 방향성도 생각해보았습니다.
- ▶ 하지만 만약 온라인 상에서 정말로 플레이어가 “솔로 플레이”인 줄 알았지만, 사실은 “멀티 플레이”인 게임을 만들기 위해서는 조금 더 많은 부분을 고려해볼 필요가 있을 것 같습니다.
- ▶ 그것과는 별개로 다양한 시스템을 직접 개발해보면서 어떤 부분이 필요하고, 어떤 변수를 제어해야 하는지 알아볼 수 있었던 좋은 프로젝트였던 거 같습니다.