

Deserted House

Deserted House 게임은 영화 폐가에서 얻은 아이디어와 현재 상황을 포함 시켜서 제작하게 된 게임입니다.



핵심 요소

아이디어

폐가라는 단어와 의미로 공포스러운 분위기와 폐쇄적인 분위기를 연출

디자인

깔끔한 맵 디자인을 위해 필요가 없는 부분은 외부 툴을 이용해 이미지 편집

코드

효율적이진 못하나 게임 개발 일정에 맞으며 개발자 실력에 맞게 전체적인 코드 구현

그래픽

깔끔한 맵 디자인을 위해 필요가 없는 부분은 외부 툴을 이용해 이미지 편집

사운드

게임 콘셉트에 맞게 공포스러운 분위기를 만들기 위해 적절한 BGM 선정

테스트

반복적인 테스트를 통해 게임 진행에 큰 영향을 주는 버그를 수정 및 개선

게임 개발 도구와 기술

Visual Studio 2022

현재 배우고 있는 VS 2022 사용

After Effect

맵 구현을 위해 보다 편리함을
제공해 주는 외부툴 사용

포토샵

캐릭터 및 전체적인 디자인

게임 개발

1

게임 장르 선택

공포 게임과 RPG 요소를 좋아하는 특성상 두 특성을 합치게 된 공포 탈출 게임을 만들게 되었습니다.

2

스토리 및 캐릭터 설계

게임 콘셉트와 맞는 특별한 캐릭터와 수집 아이템을 수집해가며 이어가는 스토리

3

개발 및 테스트

장기간 개발 및 테스트를 통해 게임 진행에 영향을 주는 버그를 발견 및 수정

개발 중 고민사항

1 실력 부족

만들고 싶은 시스템은 많지만 실력이 부족하여 개발에 어려움

2 디자인

플레이어 시점에서 직관적이고 심플하게 보일 수 있도록 디자인 고민

3 효율성

효율적인 코드 및 최적화

시작 화면



공포 분위기 연출

공포스러운 배경과 조화롭게 이루어지는 이미지를 활용해 분위기 연출

```
if (FirstDHouse::FirstDHouse())  
{  
    stage1 = new Stage1();  
    stage2 = new Stage2();  
    hidden1 = new Hidden1();  
  
    background = new Quad(L"Textures/DHouseTexture/FirstBackground.png");  
    background->Pos() = { CENTER_X, CENTER_Y };  
    background->UpdateWorld();  
  
    start_Btn = new Button(L"Textures/DHouseTexture/start_Btn.png");  
    start_Btn->Pos() = { CENTER_X, 100 };  
  
    esterEgg_Btn = new Button(L"Textures/DHouseTexture/Easter_Egg.png");  
    esterEgg_Btn->Pos() = { 100, 50 };  
  
    // option_Btn = new Button(L"Textures/DHouseTexture/Option_btn.png");  
    // option_Btn->Pos() = { CENTER_X, 100 };  
}
```

간단한 코드 구조

반복적인 이미지 출력 코드를 이용한 간단하면서도 최적화된 코드

```
if (start_Btn->isDownCheck())  
{  
    SceneManager::GetInstance()->ChangeScene("Stage1"); // 스테이지 1 씬 등록  
  
    PLAYERDATA->objectA = false;  
    PLAYERDATA->objectB = false;  
    PLAYERDATA->objectC = false;  
    PLAYERDATA->objectD = false;  
    PLAYERDATA->objectE = false;  
    hidden1->count = 0;  
  
    Audio::GetInstance()->Stop("FirstBGM");  
}  
else if (esterEgg_Btn->isDownCheck())  
{  
    Audio::GetInstance()->Stop("FirstBGM");  
    Audio::GetInstance()->Add("LostArk", "Sounds/DHouseSound/LOSTARK_Soundtrack.wav");  
    Audio::GetInstance()->Play("LostArk", 10.0f);  
  
    SceneManager::GetInstance()->ChangeScene("EsterEgg"); // 이스터에그 씬 등록  
}
```

싱글턴 최적화

싱글턴 기능을 활용하여 직접적인 인스턴스를 만들지 않고 싱글턴으로 데이터 접근 가능

Deserted Hous

게임 시작

"시작하려면 클릭하세요" 버튼을 이용해 클릭이 감지되면 출력하던 씬 자체를 변경해서 게임을 시작할 수 있도록 합니다.

시작하려면 클릭하세요

Deserte

시작하려면

이스터에그

왼쪽 하단에 보이지 않는 버튼을 클릭 시 개발자가 숨겨 놓은 사진과 배경음이 출력됩니다.


```
#include "Framework.h"
```

```
FirstDHouse::FirstDHouse()
```

```
{
```

```
    stage1 = new Stage1();
```

```
    stage2 = new Stage2();
```

```
    hidden1 = new Hidden1();
```

```
    background = new Quad(L"Textures/DHouseTex
```

```
    background->Pos() = { CENTER_X, CENTER_Y };
```

```
    background->UpdateWorld();
```

```
    start_Btn = new Button(L"Textures/DHouseTe
```

```
    start_Btn->Pos() = { CENTER_X, 100 };
```

```
    esterEgg_Btn = new Button(L"Textures/DHous
```

```
    esterEgg_Btn->Pos() = { 100, 50 };
```

```
    // option_Btn = new Button(L"Textures/DHou
```

```
    // option_Btn->Pos() = { CENTER_X, 100 };
```

```
}
```

간단한 코드 구조

이미지 출력을 위한 Quad 함수를 이용해 이미지로 공포스러운 분위기를 연출

```

FirstDHouse::Update()

start_Btn->Update();
esterEgg_Btn->Update();
// option_Btn->Update();

if (KEY_DOWN(VK_LBUTTON))
{
    if (start_Btn->isDownCheck)
    {
        SceneManager::Get()->ChangeScene("Stage1"); // 스테이지 1 씬 등록

        PLAYERDATA->objectA = false;
        PLAYERDATA->objectB = false;
        PLAYERDATA->objectC = false;
        PLAYERDATA->objectD = false;
        PLAYERDATA->objectE = false;
        hidden1->count = 0;

        Audio::Get()->Stop("FirstBGM");
    }
    else if (esterEgg_Btn->isDownCheck)
    {
        Audio::Get()->Stop("FirstBGM");
        Audio::Get()->Add("LostArk", "Sounds/DHouseSound/LOSTARK_Soundtrack");
        Audio::Get()->Play("LostArk", 10.0f);

        SceneManager::Get()->ChangeScene("EsterEgg"); // 이스터에그 씬 등록
    }
    // else if (option_Btn->isDownCheck)
    // {
    //     SceneManager::Get()->ChangeScene(""); // 옵션 씬 등록
    // }
}

```

싱글턴 최적화

클래스를 포인터 인스턴스로 생성하지 않고 싱글턴을 이용해 최적화하는 방법으로 코드 구현

스테이지 기능



이미지를 활용한 맵

코드를 이용해 개별로 만든 맵
이 아닌 통으로 된 이미지를 활
용해 게임 최적화



오브젝트 상호작용

상호작용 키를 이용한 오브젝트
활성화



콜라이더를 활용한 충돌

콜라이더를 활용해 벽과의 충돌
을 구현



이미지를 활용한 맵

코드로 하나하나 구현한 것이 아닌 이미지 편집을 이용해 전체적인 맵을 구현함으로써 게임 최적화에 직접적인 기여

: 145



오브젝트 상호작용

게임상에서 지정한 상호작용 키(G)를 이용해 맵 곳곳에 숨겨져있는 오브젝트와 상호작용하여 스토리 진행 가능



콜라이더를 활용한 충돌

콜라이더 클래스를 활용해 캐릭터가 벽이랑 부딪치는 기능을 구현하여 맵 이동의 제약을 걸 수 있다

엔딩



엔딩의 차이점

수집한 수집품에 따라 달라지는
엔딩



조건별 맵 활성화

수집한 수집품에 따라 이동하는
맵이 달라지는 기능



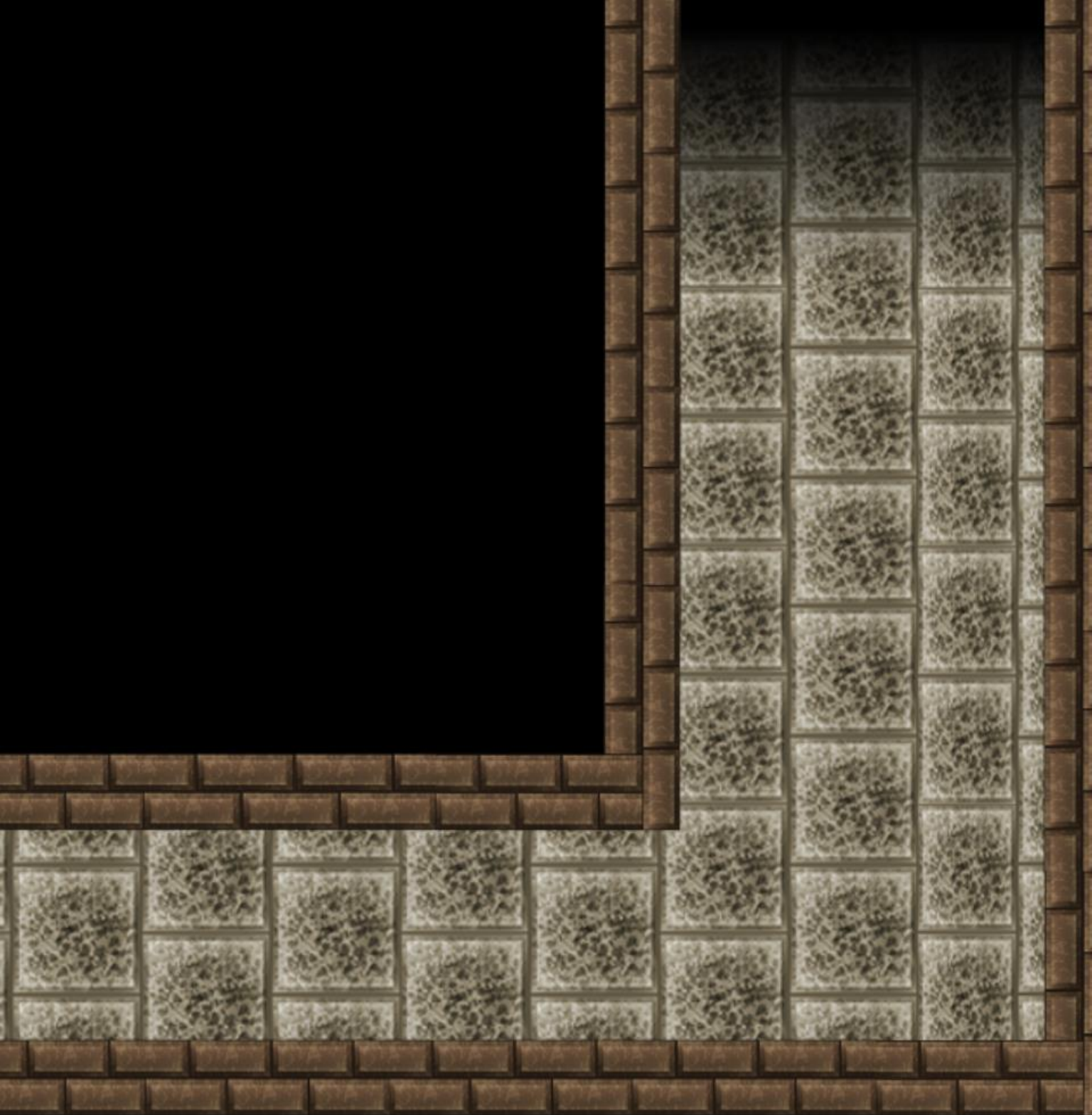
히든 엔딩

수집품을 전부 수집 했을 경우
에만, 볼 수 있는 히든 엔딩

CLEAR

엔딩의 차이점

수집품의 수집 현황에 따라 "일반 클리어"가 될 지 "히든 클리어"가 될지 정해지며 "흰색"은 일반 클리어 "노란색"은 히든 클리어 구분



조건별 맵 활성화

수집품 5개를 전부 수집할 시 히든 클리어가 가능한 맵으로 이동 가능하며, 1개라도 수집을 못 할 시, 일반 클리어만 가능한 맵으로 이동 가능하게 구현

87 #02

On August 07, 2023,
WinAPI will finally start after completing
the foundation of the C++ language among
computer languages.

Everyone is scared and excited,
but on the other hand,
it was an opportunity to know that
we have grown to this extent

820 #03

On August 21, 2023,
it was finally the first portfolio presentation
day. I hope all the projects
that everyone worked hard to complete
are completed well

700 #05

I'm about to leave
my last note on November 23, 2023.
After estimating and calculating all the dates,
it is rounded up to 700.
Seven used to mean a lucky number.
I hope everyone in the class will be
happy now and in the future

히든 엔딩

수집품 5개를 전부 수집할 시 히든 엔딩이 출력되며 히
든 엔딩은 플레이어가 게임하는 중 수집만 모든 수집품
을 한 눈에 볼 수 있도록 화면에 출력