

## 제 목 : Tower Of Angra

김 홍 주, 임 호 혁

한국산업기술대학교 게임공학부

hongju940809@naver.com, ghgur91@naver.com

### 요 약

2018 년 Goty 의 많은 수상작들의 게임 장르를 고려하여 RPG 게임을 구현하기 위해 Unreal Engine 4 를 사용하여 졸업연구를 진행하였으며 더 나아가 MMORPG 로 확장시켜 다중 멀티플레이로 사용자들 모두 만족시킬 수 있는 게임을 제작하였다. 우리는 제작과정동안 많은 우여곡절이 있었지만 포기하지 않고 개발 기술들을 습득하여 게임 속에 활용하였고 많은 것들을 느끼며 배웠다. 이 기록으로 인해 더욱 더 발전된 게임을 개발하기 위해 멈추지 않고 나아가고자 노력하겠다.

무기 교체를 할 수 있으며 최종적으로

클리어 시간과 점수를 변환하여

경쟁적인 요소를 같이 넣어 동시에

즐길 수 있도록 기획.

### 1. 서 론 1)

#### 게임의 제작 목적

- 3 차원 RPG 게임 제작
- 서버 연결하여 멀티게임 제작
- 오픈월드방식으로 넓은 맵 제작
- 캐릭터의 다양한 애니메이션 및 Skeletal Mesh 를 사용한 모델 데이터의 이해
- 스테이지에 적용할 다양한 라이팅 효과 제작

#### 2) 게임 기획 의도

- 여러 플레이어들이 타워 안으로 들어가 협업을 하여 몬스터들을 제거하고 아이템을 습득하여 점수 및

#### 3) 게임 주요 특성

- 싱글모드와 멀티모드를 각자 즐길 수 있으며 각 캐릭터는 두 가지 종류로 검과 도끼의 무기가 있으며 다양한 스킬 및 콤보 공격을 함.

#### 4)플랫폼

- 운영체제 : Windows 10
- 프로그램 : Unreal Engine 4, Visual Studio 2017

- 언어 : C/C++

#### 5)장르

- 3D 어드벤처 RPG 게임

#### 6) 게임의 스토리

- 아주 먼 옛날 왕국을 지키는  
정령들과 악마들이 존재하는  
세상. 욕심 많고 평화로운 세상이  
따분한 악마들 지배하는 마왕.  
마왕은 악마들과 함께 평화로운  
왕국을 습격하여 공주를 납치하고  
정령들의 약점을 이용하여 자신의  
탐을 지키도록 만든다. 며칠 동안  
의식을 잃은 왕은 정신을  
차리지만 소중한 기억들을 마왕이  
빼앗아갔다. 왕은 정예의  
전사들을 불러 왕국과 납치당한  
공주, 조종당하고 있는 정령, 왕의  
기억을 찾아 달라고 부탁을 한다.  
정예의 전사들은 평화로운 왕국을  
다시 되찾기 위해 마왕이 살고  
있는 탑으로 향하여 게임이  
진행된다.

## 2. 개발 내용

### 1) 개발한 게임의 내용

#### - 클라이언트

3 차원 RPG 게임으로 맵 전체적인  
이미지가 Tower 형식이며 그 안에서  
캐릭터와 몬스 터 및 여러 객체들이  
존재한다. 각 플레이 할 캐릭터들의  
이미지가 다르며 공격력 및 스킬이  
다르다. 몬스터의 종류는 총 4 가지이며  
체력, 공격력, 속도, 점수가 각각 다르다.  
또한 게임 진행 중 각 캐릭터마다

무기를 교체할 수 있으며 교체 시  
공격력 이 올라간다. 또한 멀티플레이를  
할 때 스 테이지 안에서는 협력을 하여  
몬스터를 잡 지만 게임을 클리어할 경우  
개인적으로 아 이템 및 몬스터 잡은  
점수를 계산하여 경 쟁을 하도록  
개발했다.

#### - 서버

C++ IOCP 모델을 이용한 멀티 스레드  
서버로 클라이언트와 서버는  
winsock 으로 소켓 통신한다. Main  
IOCP 에서 모든 패 킷들의 처리가  
이루어지고 Main IOCP 에 는 세 종류의  
스레드가 있다. 첫 번째 스레드:

Input/Output Thread : IOCP 를 생성하  
고 접속 요청을 하는 클라이언트로부터  
접 속을 받는 스레드이다. 두 번째  
스레드:

Worker Thread : IOCP 큐에서 작업을  
받아서 논리적인 일을 수행하는  
스레드입니 다. 예를 들어 플레이어들이  
실시간으로 보 내는 좌표 값들이나  
플레이어들과 몬스터들 의 충돌이  
일어났다거나 했을 시 서버에서 받아서  
일을 수행하는 역할을 하고 있다. 세  
번째 스레드:

Monster Thread : 게임월드 내의 몬스  
터들을 관리하는 스레드이다.

### 2) 게임 전체 흐름도

#### - 싱글 모드

- ① 캐릭터 'Warrior'와 'Axe'중 하나를  
선택한다.
- ② 맵 전체적으로  
오픈월드이므로 제한된 공간을 제외한  
모든 공간을 갈 수 있으며 몬스터들과 싸  
우거나 도망 다니며 게임을 진행한다. ③

타워의 꼭대기 층에 올라가게 되면 하나의 스테이지가 끝나게 된다.

#### - 멀티 모드

① 먼저 IP 주소를 입력 후 캐릭터 'Warrior' 와 'Axe' 중 하나를 선택한다.

② 맵 전체적으로 오픈월드이므로 제한된 공간을 제외한 모든 공간을 갈 수 있으며 협력을 하여 몬스터들과 싸우거나 각자 플레이를 할 수 있다. ③ 플레이어 중 한명이 먼저 타워의 꼭대기 층에 올라가게 되면 하나의 스테이지가 끝나게 된다.

#### 3) 개발한 방법론

- 클라이언트 - ① Title 및 게임모드 : 블루프린트 클래스 와 위젯 블루프린트를 만들어 제작하였고

C++ 클래스를 만들어 C++ 와 Blueprint 간의 상호작용을 했다.

② Map 과 Lighting : Map 을 오픈월드처럼 자연스러운 Map 변환을 위해 Level Changer Actor 클래스를 만들어 특정 위치에 가게 되면 다음 Map 을 로드 시킨다. 또한 게임이 Tower 안에서 진행하기 때문에 빛을 만드는 촛불 객체와 Light 가 많이 필요해 빌드를 할 때 오래 걸리면서 많은 메모리가 차지했다. 이를 해결하기 위해 모형으로 촛불 객체를 만들어 실제로 PointLight 를 Static 으로 만들었으며 빛과 그림자를 표현하기 위해 RectLight 를 따로 만들어 Lighting 효과 표현했다.

③ 캐릭터 : C++ 로 구현하였으며 두 종류의 캐릭터로 TowerofAngraCharacter 를 상속받아 중복되는 코드(움직임, 콤보 공격, 공통 스킬)를 작성하고 각자 특성 스킬 및 애니메이션을 따로 제작했다. 공격 애니메이션과 스킬 애니메이션은 애니메이션 몽 타주를 만들고 C++ 와 상호작용을 하도록 구현했다. 또한 스킬을 사용하면 카메라가 흔들리는 효과를 주기위해 Camera Shake Class 를 제작했다.

④ 몬스터 : 캐릭터와 마찬가지로 C++ 로 구현하면서 HMonster 를 상속받았으며 AI 는 Behavior Tree 를 이용하였으며 구현 방법으로는 C++ 를 사용하여 Behavior

Tree 를 사용하도록 Targer, Patrol, Attack Range 등 다양하게 AI 를 개발했다.

⑤ UI : UMG 를 이용하여 각 캐릭터 및 몬스터들의 HP, MP 를 표현하였으며 기존에 있는 Bar Tool 이 아닌 이미지를 사용하여 퀄리티가 있도록 제작했다. ⑥

객체 : 아이템 객체와 보물 상자 객체 발판 객체 문 객체 등 C++ 를 사용하였으며 OverlapVolume 을 사용하여 정해진 공간에 플레이어가 들어오면 Overlap 이 실행 되도록 제작했다. 또한 객체가 플레이어와 충돌하는 순간 사라지도록하기 위해 Trace Channels 과 Preset 을 설정하였으며 충돌을 하게 되면 Effect 효과 후 객체를 삭제 한다.

⑦ 서버와 동기화 : 서버로 플레이어 상태 와 몬스터의 정보를 받기위해 각 객체들을 Enum 값과 전역 변수를 사용했다.

- 서버 - 게임 개발 방법론은 게임의 제작이나 개발,

게임의 테스트나 관리를 위하여 사용되는 방법으로써 게임 제작 시 과정이나 절차가 명확 하게 표현 되어야 게임 제작 과정의 현 주소 를 알 수 있고 앞으로의 작업 과정을 쉽게 파악가능 했다.

게임 개발 과정으로는 게임 개발을 위해 게임 기획이나 시장 조사 등을 진행 하고 제작 에 들어서게 됐다. 제작과정으로는 클라이언트와 서버 코딩 및 동기화 과정을 진행 했다. 그 뒤 개발이 마친 게임이 버그가 있는지 테스트를 진행 하였고 실제 플레이어들이 플레이 하였을 때 오류가 나지 않도록 작업을 진행 했다.

### 3) 게임 진행 방법

- 조작 방법 : 키보드, 마우스 상하좌우 - W, S, A, D 공통 스킬 - Q 카메라 이동 - 마우스 무빙 공격 - 마우스 오른쪽 버튼 (마우스 오른쪽 버튼을 연타를 누를 시 콤보 공격을 함.)

스킬 - 마우스 왼쪽 버튼

- 게임 진행 플레이어가 Tower 의 특정 지점에서 시작되 며 여러 종류들의 몬스터들 제거하면서 게 임을 진행한다. 진행 중 물약과 보물 상자 및 무기들을 습득하고 Tower 꼭대기에 올라 가게 되면 게임 클리어를 할 수 있다.

### 4) 게임의 특성

Tower Of Angra 는 액션 RPG 게임으로써 유저들이 게임월드 에 들어와 다양한 동작을

실시간으로 조작하여 이야기를 풀어가는 게임 으로 기획하고 제작했다. 이러한 액션 게임의 주된 요소는 원초적인 반사 신경과 조작능력 으로 적을 공격하면서 플레이어들은 이런 과 정을 통해 재미를 느낀다. 우리의 게임은 재 미있고 직관적인 게임을 만들기 위해 노력했 으며 적을 제거하되 “같은 플레이어보다 많은 적을 처치하라” 라는 목표를 가지고 만들었고 직관적인 게임을 만듦으로써 진입장벽을 낮추 고 누구나 쉽게 게임을 접할 수 있는 3 차원 게임이다. 덕분에 라이트 게이머부터 하드한 게이머까지 모두가 즐길 수 있도록 제작을 하 였고 한 판을 플레이 하는데 10 분이 걸리지 않도록 제작했다. 그리고 플레이어가 몬스터 를 일반 및 스킬을 이용한 공격을 했을 때는 화려한 이펙트가 나오도록 하고 그로인해 타 격감이나 속도감 등 플레이어가 액션을 더욱 쾌감을 느끼도록 받아들이도록 했다.

## 3. 결 론

### 1) 개발된 작품의 장단점

- 클라이언트 -

① 장점 : 언리얼 엔진을 사용하면서 블루 프린트를 최소화하였고 C++ 를 위주로 사 용하여 게임제작에 임했다. Tower 안에서 게임을 진행하기 때문에 Lighting 효과를 극대화 했다. 공격 모션 및 스킬을 사용할 때 Effect 효과를 줘서 시각적인 극대화를 했다.

② 단점 : 맵에 대한 기획을 확실하게 하지 못하여 처음 진행하는 플레이어들에게는 어려움을 느낄 수 있다. 또한 다양한 사운드를 사용하지 못한 아쉬움과 보스구현을 기한 내에 못한 점이 있다. - 서버 (김 홍 주) ① 장점 : 패킷의 정보를 직렬 화 및 역 직 렬

화하여 stream 으로 통신하는데 서버와 클라이언트들 간의 패킷 송수신을 간편하게 했다는 점이다.

② 단점 : DB 연결을 하지 못한 점  
아쉽다. DB 를 게임서버와 연결해서 로그인 시스템도 구현을 했어야 하고 정보들을 저장할 못한 점이 아쉽 게 느껴졌다. 그 다음 최적화를 생각하고 작성 한 코드가 아니라는 점이다. 물론 게임서버의 속도는 DB 연결 및 패킷 송수신이 가장 큰 요소 를 차지하지만 코드를 작성할 때 최적화를 많이 생각해서 작성한 코드는 아니다.

## 2) 향후 추가 개발

- 클라이언트 (임 호 혁)

- ① 스테이지 진행에 대한 방향 UI 제작
- ② 몬스터 레벨 디자인 수정
- ③ 보스구현 및 다양한 AI 제작
- ④ 다양한 사운드를 적용
- ⑤ 필요 없는 Class 제거 및 중복되는 함수 재정리

- 서버 -

- ① 게임서버와 DB 를 연결해서 로그인 및 회원가입 시스템을 추가 개발예정.
- ② 게임 플레이시 발생하는 재화, 플레이어의 정보 등 여러 가지 데이터를 보안 처리 하는걸 추가 개발예정
- ③ AES 알고리즘을 학습하고 적용

## 4. 부 록

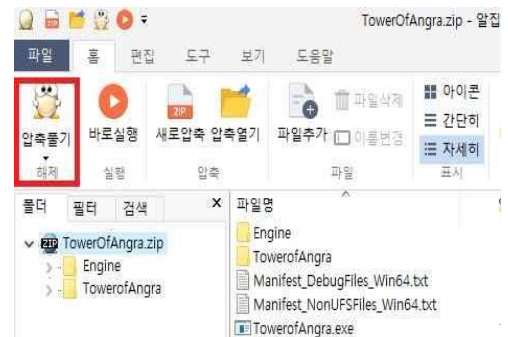
### 1) 게임 실행 환경:

- 권장 하드웨어

- ① 운영체제 - Windows 10 (64bit)
- ② 프로세서 - 쿼드 코어 Intel 또는 AMD,  
2.5G  
Hz 이상
- ③ 메모리 - 8GB RAM
- ④ 비디오 카드 / DirectX 버전  
- DirectX11 또는 DirectX12 호환 그래픽카드

### 2) 게임 설치

① TowerOfAngra.zip 을 풀어준다.



[그림 1]

② TowerOfAngra 파일에서 TowerofAngra 응용프로그램을 실행한다.



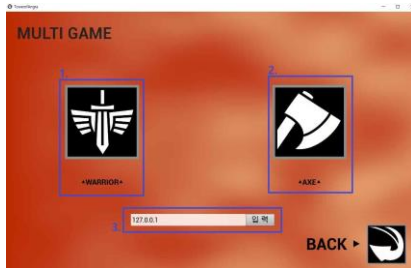
[그림 2]

### 3) 게임 진행



[그림 3]

가) [그림 3] 시작화면으로 SingleGame 과 MultiGame 을 정하여 진행한다.



[그림 4]

나) [그림 4] IP 주소입력 후 캐릭터를 선택하면 게임이 시작된다. (싱글모드는 캐릭터를 선택하면 게임이 시작된다.)



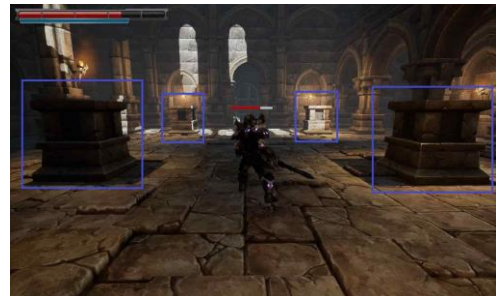
[그림 5]

다) [그림 5] 멀티 플레이로 양쪽의 플레 이어들이 접속하여 게임을 진행한다.



[그림 6]

라) [그림 6] 몬스터들과 액션장면이다.



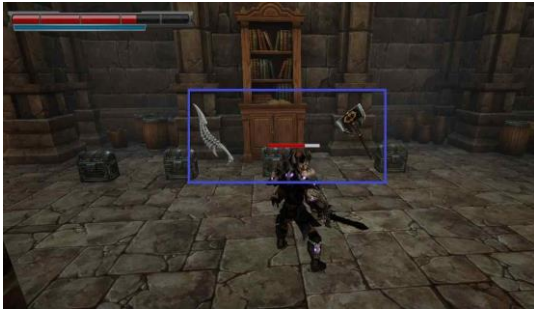
[그림 7]

마) [그림 7] 4 개의 발판을 점프하여 밟아 야 앞의 문이 열리며 진행된다.



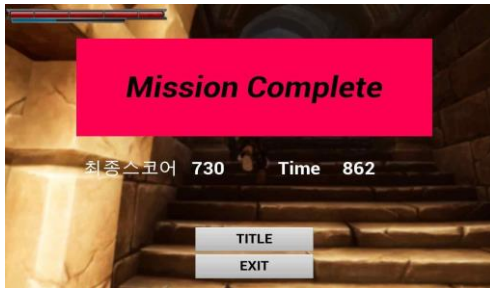
[그림 8]

바) [그림 8] 보물 상자 및 물약들을 먹고 진행한다.



[그림 9]

사) [그림 9] 캐릭터의 속성에 맞는 무기를 얻을 수 있다.



[그림 10]

아) [그림 10] 게임을 클리어하게 되면 나 오는 장면이다.

#### 4) 문제 해결

- 클라이언트 -

##### ① PlayerCameraShake 구현

문제점

Player 가 스킬 공격을 사용할 때 이펙트와 사운드만으로는 밋밋한 느낌.

해결

```
UHPPlayerCameraShake::UHPPlayerCameraShake()
{
    OscillationDuration = 0.25f;
    OscillationBlendInTime = 0.05f;
    OscillationBlendOutTime = 0.05f;

    RotOscillation.Pitch.Amplitude = FMath::RandRange(5.f, 10.f);
    RotOscillation.Pitch.Frequency = FMath::RandRange(25.f, 35.f);

    RotOscillation.Yaw.Amplitude = FMath::RandRange(5.f, 10.f);
    RotOscillation.Yaw.Frequency = FMath::RandRange(25.f, 35.f);
}
```

[그림 11]

Pitch 와 Yaw 의 흔들림을 주기 위해 RandRange 의 범위의 랜덤 값을 주어 진동 을 표현.

##### ② 다수의 충돌 구현 문제점

스킬 공격 범위에 있는 몬스터들에게 데미지를 주는 과정에서 대다수에게 데미지가 가지 않고 특정 몬스터에게만 데미지를 입는 문제

해결

```
void AHPCharacter::SkillCheck()
{
    // 공격범위
    float AttackRadius = 500.f;

    TArray<OverlapResult> OverlapResults;
    FCollisionQueryParams Params(NAME_None, false, this); // Collision 조건

    bool bResult = GetWorld()->OverlapMultiByChannel(
        OverlapResults,
        GetActorLocation(),
        FQuat::Identity,
        ECC_GameTraceChannel2,
        FCollisionShape::MakeSphere(AttackRadius),
        Params
    );

    if (bResult)
    {
        for (auto OverlapResult : OverlapResults)
        {
            AHPMonster* Monster = Cast<AHPMonster>(OverlapResult.GetActor());

            if (Monster)
            {
                FDamageEvent DamageEvent;
                Monster->Damage(1.f);
                if (CurrentDamageMode == EDamageMode::MULTI_GAME)
                {
                    Monster->MonsterDamageEffect(FSkillPower);
                }
                else
                {
                    OverlapResult.Actor->TakeDamage(FSkillPower, DamageEvent, GetController(), this);
                }
            }
        }
    }
}
```

[그림 12]

(1) OverlapMultiByChannel 함수를 사용하여 공격범위 안에 있는 Character 정보를 담음.

(2) 범위 안에 있는 몬스터들을 공격하기 위해 For 문을 사용하였고 Character 정보 중 몬스터만 분별하여 데미지를 입도록 함.



1) 타깃을 찾을 시트리 조건

2) 타깃을 못 찾을 시트리 조건

Behavior Tree 를 사용하여 Player 가 정해진 범위에 없으면 몬스터들은 랜덤 값으로 움직이며 Player 를 찾으러 다닌다. 범위 안에 들어오게 되면 타겟으로 정하여 범위가 벗어나지 않을 경우 계속 쫓아 다니며 공격.

## 문제점

C++를 어떻게 구현할 것인지 문제.

## 해결

[그림 6]

(2) 몬스터 중 플레이어를 찾은 시  
타깃으로 잡고 플레이어를 따라가도록  
구현. 추가적으로 Player 를 타깃으로  
잡게 되면 몬스터 HP 현황이  
플레이어에게 보여 짐.

해결



Nagle 알고리즘을 중단했다. 그 이유는 Nagle 알고리즘의 원리는 ACK 를 받은 다음에 데이터를 보내고 ACK 를 받을 때까지 출력버퍼의 데이터를 저장하였다가 ACK 를 받으면 버퍼의 데이터를 모두 패킷으로 만들어 보낸다는 것이다. 또한 Nagle 알고리즘은 가능하면 조금씩 여러 번 보내지 말고 한 번에 많이 보내라는 원칙을 기반으로 하고 있는데 네트워크의 전송량이나 부하보다는 빠른 응답속도를 원했기 때문에 TCP\_NODELAY 라는 옵션을 사용하여 Nagle 알고리즘을 중단하고 개발을 진행했다.