

Unity 3D 엔진을 이용한 모바일 RPG 게임 개발

김호렬*, 유종호*, 한창민*, 안성욱*, 김수균^o
 배재대학교, 게임공학과^o
 배재대학교, 게임공학과*
 e-mail: kimsk@pcu.ac.kr^o

Development of Mobile RPG using Unity 3D Engine

Horyul Kim*, Jongho Yoo*, Changmin Han*, Syoungog An*, Soo Kyun Kim^o
 Dept. of Game Engineering, Paichai University^o
 Dept. of Game Engineering, Paichai University*

● 요약 ●

게임 시장에 높은 점유율을 보이고 있는 추세에 따라 모바일 게임 제작에 많이 사용하고 있는 Unity 3D 게임 엔진을 사용하였다. 모바일 RPG인 ‘몬스터 길들이기’와 ‘Hundred Soul’ 이라는 게임에서 감명 받아 이러한 아이디어를 바탕으로 Unity 3D 게임 엔진의 여러 기능과 접목하여, 다양한 기능 및 효과들을 구현하였고, 그에 따른 개발 방법을 설명한다.

키워드: 유니티 3D(Unity 3D), 게임 엔진(Game Engine), 모바일 RPG(Mobile RPG), 오브젝트 풀 (Object Pool), 유한 상태 기계(Finite State Machine)

I. Introduction

RPG(Role-Playing Game)는 플레이어가 게임 속 가상 세계에서 자신에게 할당된 캐릭터를 조작하고 가상의 상황에서 위기를 극복하여 목표를 달성하는 게임의 일종이다. PC 환경과 다른 모바일 플랫폼을 고려하여 가상 조이스틱과 버튼을 통해 캐릭터를 조작한다. 최근 모바일 게임 동향을 살펴보면 RPG 장르의 게임들의 매출 순위가 높다. 그림 1에서의 같이 플레이스토어의 매출 순위의 상위권이 RPG 인 것을 알 수 있다.



Fig. 1. Google PlayStore Ranking : Top Paid

본 논문에서는 Unity 3D 게임 엔진을 사용하여 모바일 RPG 개발하는 방법을 설명한다.

II. The Proposed Scheme

1. Object Pooling

오브젝트 풀링(Object Pooling)[1]은 그림 2와 같이 객체를 미리 생성하여 배열이나 리스트에 저장해 객체가 필요할 때 마다 활성화하고 필요하지 않을 때 비활성화한다. 이는 런 타임 중 garbage 컬렉션이 발생하는 것을 방지할 수 있고 객체가 필요할 때 마다 생성을 하지 않아도 되기 때문에 게임을 최적화하는데 중요한 요소이다.

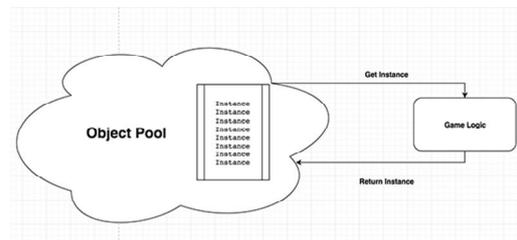


Fig. 2. Object Pooling

2. Finite State Machine

유한 상태 기계(Finite-State Machine)[2]는 그림 3과 같이 유한한 개수의 상태(State)를 가지고 있는 기계라는 의미로, 기계는 한 개의 상태만 유지할 수 있고 그 상태에 맞는 특별한 이벤트나 상황이 되었을 때 다른 상태로 전이하여 자신의 일을 수행하는 특징을 가진다.

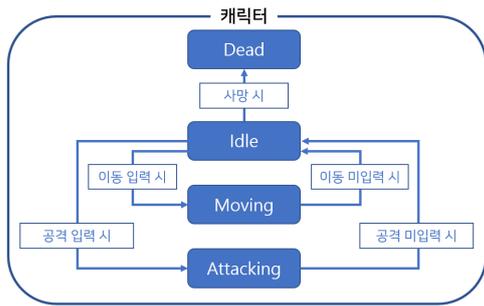


Fig. 3. Character FSM

3. Observer Pattern

옵저버 패턴(Observer Pattern)은 하나 이상의 객체를 관찰 대상이 되는 객체에 등록시켜 관찰 대상에서 발생되는 이벤트를 받아 처리하는 패턴이다.

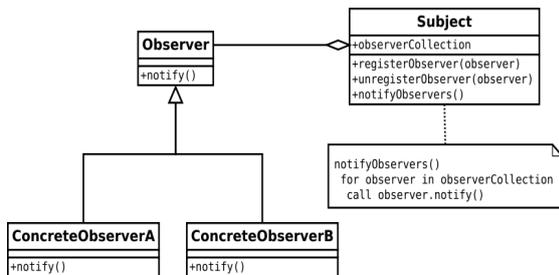


Fig. 4. Observer Pattern UML Diagram

그림 4와 같이 Subject라는 객체가 Observer를 가지고 있어 이벤트 발생 시 Observer내의 메소드를 호출해 관리하는 방식이다. 인 게임 내에서는 특정 객체들이 옵저버 형태의 미션을 가지고 있어 이벤트가 발생 시 각 옵저버로 콜백을 보내 미션 진행도를 변경해준다. 이를 통해 클래스 간의 의존성이 낮아져 유연한 코딩을 할 수 있게 된다[3].

4. Swapping

인-게임내의 스와핑 기능을 통해 플레이어의 캐릭터를 변경하며 플레이 가능하며 이를 통해, 사용자에게 난이도 있는 조작감을 요구하여 흥미를 유발하도록 한다.



Fig. 5. Character Swapping

그림 5와 같이 특정 캐릭터 버튼 터치 시 해당 캐릭터로 카메라가 이동하게 된다.

5. Character Skill

캐릭터들의 스킬을 캐릭터의 고유 기술이 아닌 [그림 6]과 같이 아이템의 고유 기술로 개발을 하여 아이템에 따른 스킬 변화를 구현하였다. 이러한 기능의 자유도를 뒷받침으로 사용자에게 전략적 요소로 하여금 흥미를 유발하도록 한다.



Fig. 6. Skill Changing by Items

6. Character Information

캐릭터 정보 UI는 모든 캐릭터 정보를 가지고 있는 CharacterManager를 통해 참조해 캐릭터 정보를 관리해 데이터를 가져다 UGUI의 Text와 Image로 정보를 출력해준다. 장비 착용은 인벤토리 데이터를 참조해 슬롯에 맞는 카테고리를 적용해 슬롯에 맞는 무기를 출력해주고 무기 슬롯(Button)을 클릭 시 슬롯에 들어있는 아이템을 인벤토리로부터 꺼내 캐릭터 정보로 넘겨주는 방식으로 구현되어있다.

III. Conclusions

본 논문에서는 Unity 3D 게임 엔진을 이용하여 모바일 RPG를 개발하는데 필요한 기술과 아이디어를 설명하였다. 이러한 기술들을 사용하여 모바일에 최적화된 게임 환경을 구축할 수 있고, 앞서 설명한 아이디어를 통해 사용자에게 더 나은 몰입감 및 흥미를 제공할 수 있다.

본 논문에서 설명한 기술들은 게임 개발자가 모바일 RPG를 개발하게 될 경우 효율적인 방법으로 사용이 가능하며 어떠한 게임에서도 적용될 수 있도록 설명하였다.

REFERENCES

- [1] Unity Tutorial : <https://unity3d.com/kr/learn/tutorials/topic/s/scripting/object-pooling>
- [2] Wiki Unity 3D : https://wiki.unity3d.com/index.php/Finite_State_Machine
- [3] Wikipedia Observer : https://en.wikipedia.org/wiki/Observer_pattern