[윈도우 프로그래밍] 최종 프로젝트 보고서

2020180033 장명운

2020182033 이시우

1. 게임 소개 및 특징, 게임의 규모, 게임 개발 결과

던그리드의 모작.

2D 횡스크롤 액션 게임으로 플레이어가 맵을 돌아다니며 다양한 몬스터를 물리치는 형식의 게임이다.

맵은 총 4개로 이루어져 있고 플레이어는 근거리 또는 원거리 공격을 통해 적을 처치할 수 있다. 몬스터는 종류에 따라 근거리 / 원거리 공격이 가능하고 공중에 떠있기도 한다. 몬스터의 일정 거리 이내에 플레이어가 있을 시 몬스터는 플레이어를 따라가 공격한다. 플레이어와 몬스터는 모두 hp가 0이 됐을 시 죽게 되고 플레이어가 몬스터를 모두 물리쳐야만 플레이어는 다른 맵으로 이동할 수 있게 된다.

DB 를 사용해 리소스만 있으면 맵 추가, 몬스터 추가 등이 용이하게 했다.

1. 실행 방법 / 필요한 라이브러리
2. 필요한 라이브러리 : FMOD Engine 2.01 Windows
3. 실행 방법 :
4. 키보드 : ‘A’ - 좌측 이동, ‘D’ - 우측 이동

스페이스바 – 점프

스페이스바 + ‘S’키 – 아래로 떨어지기 (아래로 내려갈 수 있는 지형에서만 가능)

1. 마우스 : 좌클릭 – 근거리 공격

휠클릭 – 원거리 공격

우클릭 – 대쉬

1. 구현한 내용, 계획했으나 구현하지 못한 내용들
2. 구현한 내용 :
   1. 주인공
3. 근거리 공격, 원거리 공격 (마우스 좌클릭, 휠 버튼 클릭)
4. 마우스 위치에 크로스헤어 존재. 주인공은 크로스헤어 방향을 보고 있고 해당 방향으로 공격한다.
5. 좌우 이동 (‘A’키, ‘D키’)
6. 점프 (스페이스바)
7. 아래로 떨어지기 (스페이스바 + ‘S’키)
8. 대쉬 (마우스 우클릭)
9. 체력 0이 되었을 시 주인공 죽음
   1. 몬스터
10. DB에 설정되어 있는 행동 확률로 아래 1~4 중 하나를 택해 자동으로 움직일 수 있게 구현

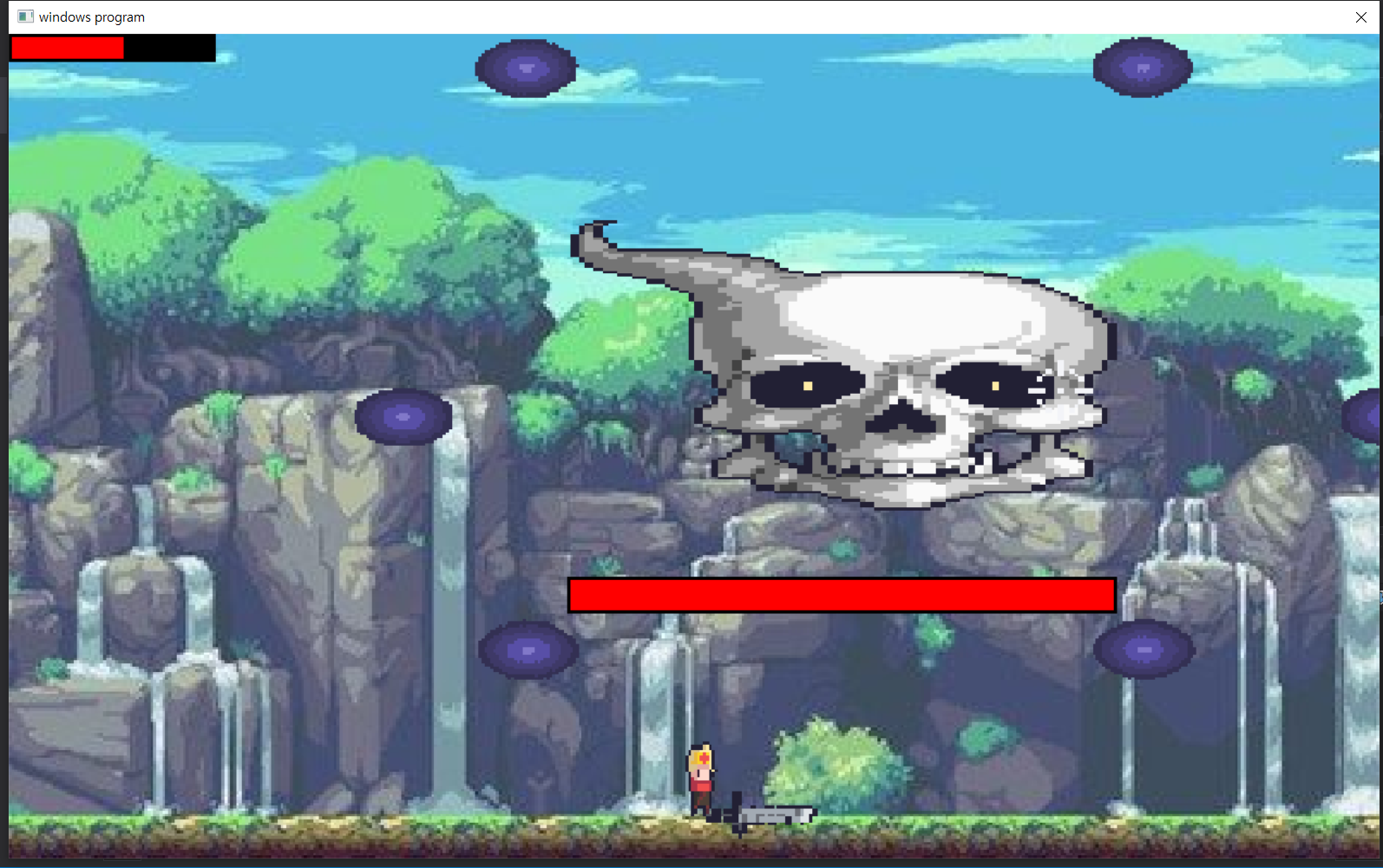
1) 정지

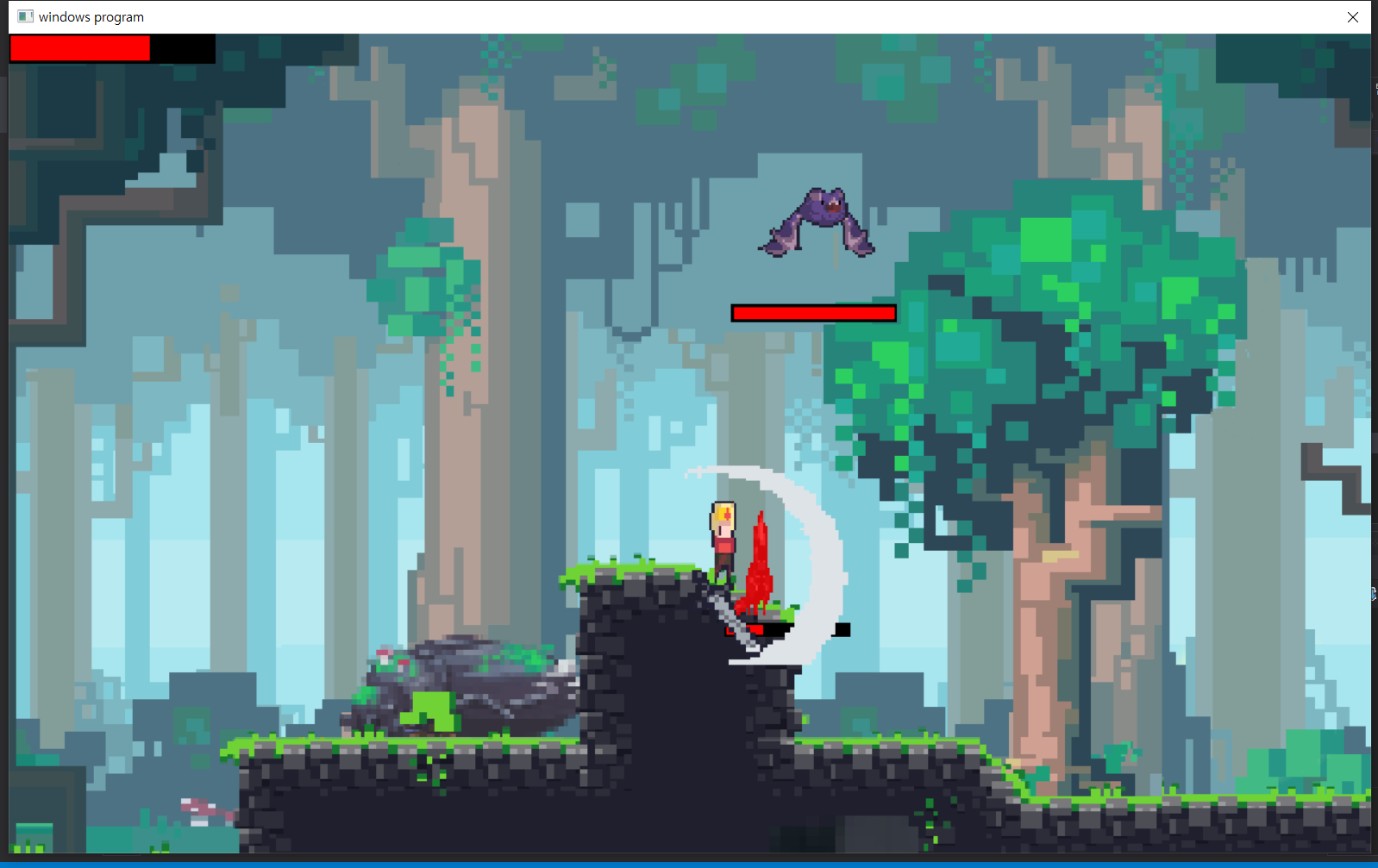
2) 플레이어를 향해 이동

3) 플레이어 반대 방향을 이동

4) 플레이어를 쫓아가 공격

1. 경우에 따라 근거리 공격 / 원거리 공격이 가능하게 구현
2. 경우에 따라 걸어 다니거나 / 떠다닐 수 있게 구현
   1. 맵
3. 4스테이지의 맵
4. 색상으로 맵의 정보를 담은 map-terrain 이미지를 통해 캐릭터의 충돌 감지를 가능하게 했다.
   1. 보스
   2. 그 외
5. 애니메이션 – DB와 AnimationManager와 Animation을 이용해 쉽고 편리하게 애니메이션 관련 이미지 / 정보들을 관리할 수 있게 했다.
6. 카메라 – 주인공의 위치에 따라 카메라 좌표를 변경해 주인공이 맵의 끝 쪽에 있는 경우가 아니면 주인공이 화면의 중앙에 오게 했다.
7. 데이터 베이스 – 맵 관련 정보 (맵의 아이디, 맵의 사이즈, 이미지, 주인공 시작 위치, 등장 몬스터 종류 / 수 등), 애니메이션 관련 정보(애니메이션 이름, 애니메이션 이미지 수, 반복 여부 등), 몬스터 정보(몬스터 이름, 사이즈, 체력, 공격력, 방어력 등)를 쉽게 관리할 수 있게 했다.
8. 이펙트 – 애니메이션을 한번 만 재생하고 사라지게 했다.
9. ImageContainer – 애니메이션 등에 사용될 이미지를 저장하고 있다.
10. HitScanner – 해당 캐릭터가 공격 당했는지를 체크하고 공격력과 방어력에 따른 데미지를 계산해 적용해준다.
11. MonsterManager – 몬스터들을 한번에 생성, Update할 수 있게 했다.
12. MonsterAI – 몬스터들이 정해진 루틴에 따라 자동으로 움직이고 공 격하게 했다.
13. 사운드 – bgm, 효과음등을 쉽게 재생할 수 있게 했다. (FMOD 사용)
14. 무기 – 크로스헤어 방향에 따라 이미지가 반전/회전 되도록 구현했다.
15. 스크린샷 / 팀원 역할 및 팀원간 개발 내용
16. 스크린샷

지도이(가) 표시된 사진

자동 생성된 설명

1. 팀원 역할 및 팀원 개발 내용
2. 장명운
3. 애니메이션
4. DB (던전, 애니메이션, 몬스터의 정보를 담고 있음)
5. 근거리 공격 / 원거리 공격
6. HitScan (피격 처리)
7. 몬스터
8. MonsterAI (몬스터 자동 이동 / 공격)
9. 보스
10. 이시우
11. 카메라
12. 크로스헤어
13. 이펙트
14. 사운드
15. 무기 회전 / 반전
16. 제작 후기
17. 장명운 : 개발 과정 속에서 불안함이 많았다. 게임 구조를 설계하는 데 들인 시간이 개발 기간의 대부분이어서, 프로젝트의 핵심 중에 하나인 회전 이미지 출력을 제외하면 발표 나흘 전까지만 해도 이 게임은 단순한 점프 게임이었다. 발표 이틀 전에야 몬스터가 추가됐고 발표 하루 전에야 공격이 가능해졌으며, 발표 당일날에 보스가 생겼으니 과연 얼마나 초조했을까. 또 눈에 안 보이는 곳에 집중한다는 게 얼마나 힘든 일인지 느껴졌다. 눈에 안 보이는 곳을 보이게 하기 위해 단위 테스트를 하는 법을 이번에 익힐 수 있었는데, 버그를 쉽게 다 잡아내고, 레고 블럭을 합치듯이 구조가 합쳐지는 게 뿌듯했다. STL의 이용까지 더불어서 개인적으로 실력이 늘은 부분도 많은 것 같다. 팀으로 작업을 하는 것도 소중한 경험이어서, git 사용법을 익히기 위해 또 적지 않은 시간을 들였고, 우리는 서로가 서로의 코드를 쉽게 쓸 수 있게 최대한 각 부분이 독립적으로 작용하도록, 함수를 사용하기 쉽도록 코드를 구성했다. 구현되길 원하는 부분을 말해서 그 부분이 구현되어서 오면, 문장 한 줄 적는 것으로 원했던 동작이 실행될 수 있다는 게 매우 놀랍고 고마웠다. 개발 막바지엔 더이상 코드의 품질을 신경 못쓰고 함수가 인자와 줄 수로 옆으로도, 아래로도 길어지게 되었는데 실제로 마감에 쫓기는 기분을 느껴봤다는 사실도 색달랐다. 그래서 초반에 짠 코드와 후반에 짠 코드를 보면 품질 차이가 느껴지는데, 이를 관찰하는 묘미도 있어서 즐겁고 뿌듯했던 프로젝트같다.
18. 이시우 : winapi도 처음이고 게임을 제작하는 것도 처음이라 어려움이 많이 느껴졌다. 내가 프로그래밍 쪽에 부족함이 있다는 걸 이번 프로젝트를 통해 확실히 느끼게 되었고 부족함을 채우기 위해 이번 방학에 더 열심히 해야겠다 생각했다.
19. 유튜브 주소

https://www.youtube.com/watch?v=XdVzdiBGZkE