

## 1. 게임 설명

- 게임 사용 방법 : 방향키로 이동하고 Shift로 달릴 수 있다. 근거리 적과 원거리 적의 공격을 피하며, TYPE(타자)로 공격하여 적을 처치한다. 일정 시간마다 소환되는 파워업 아이템과 획득하여 위기를 모면할 수 있다.
- 게임 완료 조건 : 3분 동안 버티기를 성공한다. 적 처치 수 & 오타 개수로 점수가 계산된다.
- 게임의 차별화 전략 : 기존의 단순한 마우스 클릭이나 버튼 입력 기반의 공격 방식이 아닌, 단어를 빠르고 정확하게 입력하여 적을 공격하는 타이핑 전투 시스템.
- 게임의 핵심 재미 요소 : 안전한 위치로 이동한 뒤 연달아 타이핑으로 적을 처리하는 전략, 혹은 이동과 타이핑을 동시에 수행하는 고난도 컨트롤 전략 등 플레이어의 스타일에 따라 다른 전투 방식을 선택할 수 있다. 파워업을 바로 획득할지, 아끼다가 위급할 때 획득할지 다양한 전략을 세울 수 있다.

## 2. 개발 내용

- 플레이어 개발 :
  - 이동 : 타이핑을 해야 하는 타자 게임 특성상 마우스를 사용할 수 없기에, 키보드의 움직임만으로 방향 전환을 가능하게 하려고 고민했다. 고민 끝에 가상의 축을 기준으로 동서남북으로 움직이는 플레이어 움직임을 구현하는 것이 낫다고 느꼈고, 그 가상의 축은 카메라가 바라보는 방향으로 설정했다.
  - 애니메이션 :
    - ◆ 이동 애니메이션 : Locomotion을 만들어 구현했다. 플레이어가 이동 속도에 따라 걷기 모션과 달리기 모션이 자동으로 전환되게 Blend Tree로 구현했다.
    - ◆ 피격 애니메이션 : 피격 방향을 계산하여 피격 방향에 따라 네 종류의 피격 애니메이션을 구현했다. 움직이면서 동시에 피격을 당할 때, 피격 애니메이션이 재생되느라 하체의 이동 애니메이션이 재생되지 않는 점이 부자연스러워서, 따로 Upper Body Layer를 분리하여 피격 애니메이션이 상체에만 재생되게 구현했다.
    - ◆ 마법 캐스팅 애니메이션 : 피격 애니메이션과 동일한 이유로 상체에만 재생되게 구현했다. 마법이 발사되는 손 끝이 목표 적을 향하도록 플레이어를 회전시키게 구현했다.
- 적 또는 NPC 개발 :
  - 근거리 적 : 근거리 적은 기본적인 상태머신을 가진다. 내브메쉬를 이용해 플레이어를 항상 따라다니고, 공격 사거리 이내에 플레이어가 들어오면 공격을 한다. 더 좋은 플레이 경험을 위해 공격을 시도하는 도중 마법에 피격을 당하면 공격이 취소되게 구현했다.
  - 원거리 적 : 원거리 적은 가만히 서서 일정 시간마다 플레이어를 공격한다. 공격에는 두 가지 패턴이 있다. 플레이어의 주위 일정 반경 랜덤한 위치에 공격을 하거나, 플레이어의 이동 속도 기반 1.5초 후의 예상 경로를 계산하여 그 주위 랜덤한 위치에 공격을 한다. 원거리 적이 투사체를 발사하는 것은 게임의 난도가 너무 높아질 것 같기에 일정 시간 후 범위 공격 소환으로 구현했다. 원거리 적이 벽에 너무 가깝게 소환되면 머리 위의 단어가 벽에 파묻혀 보이지 않기에, 원거리 적은 소환 범위를 조금 줄였다.
- 인터랙션 :
  - 근거리 적의 공격 애니메이션이 왼손을 휘둘러 공격을 하기에 왼손에 콜라이더를 부착시켜서 플레이어와 충돌하면 데미지를 입히게 했다. 적이 공격 중이 아닐 때도 해당 콜라이더와 충돌했을 때 데미지를 입는 불편한 플레이 경험을 없애기 위해 애니메이션 키 프레임 이벤트를 활용하여 공격 애니메이션이 본격적으로 활성화되는 시간 동안만 콜라이더를 활성화시키게 했다.

- 효과 :
  - 사용한 파티클 : 플레이어 마법 투사체, 플레이어 마법 임팩트(타격), 파워업의 오오라, 원거리 적의 마법.
  - 사용한 오디오 : BGM 두 종, 플레이어 피격 및 사망 소리, 플레이어 마법 발사 및 임팩트 소리, 게임 오버 및 승리 소리.
- 카메라 및 UI :
  - 카메라 : 마우스를 사용하지 못하는 게임 장르의 특성상 고정된 각도의 카메라가 플레이어의 움직임을 따라오게 했다.
  - UI : 메인 화면의 게임 시작, 플레이 방법, 환경 설정, 크레딧 버튼과 창을 구현했다. 인 게임에서는 플레이어의 체력, 스테미너, 점수, 남은 시간 등을 표시한다. 플레이어의 체력이 한 칸 남았을 때는 화면 주위에 빨간 경고 효과를 표시하게 했다. 파워업 아이템을 획득하면 해당 아이템의 설명이 표시되게 구현했다. 최고 점수를 PlayerPrefs 으로 저장하여 관리한다.

### 3. 최적화

- 적을 오브젝트 풀링 방식으로 관리하여 메모리 할당 및 해제를 최적화 하였고, 가비지 콜렉터의 부담을 덜어주었다.
- 처음엔 마법 효과와 피격 효과 등을 Visual Effect로 구현하였으나, Visual Effect가 WebGL에서 안정적으로 렌더링되지 않는다는 사실을 깨닫고 모든 효과들을 Particle System으로 교체했다.

### 4. 애셋 활용 (이하의 애셋 뿐만 아니라 모든 사용한 애셋 목록은 게임의 “크레딧” 창에서 확인 가능)

- 3D 모델 : [Mixamo](#) (Abe, Vampire A Lusth, Demon T), [Simple Gems and Items Ultimate Animated Customizable Pack | 3D Props | Unity Asset Store](#)
- 애니메이션 : [Mixamo](#) (Lite Magic Pack, Standing 2H Magic, Magic Locomotion, Creature NPC Pack)
- 애셋 활용에서 본인 기여 :
  - 애셋의 스크립트를 사용한 요소는 전혀 없다.
  - “적군 이동 속도 감소” 효과 파워업에 해당하는 모델이 딱히 없기에, “플레이어 이동 속도 증가” 파워업에 쓰였던 3D 모델을 Blender를 사용하여 모든 정점들을 위 아래로 뒤집어 추출하여 사용했다.
  - 근거리 적의 공격 애니메이션의 시전 시간이 길어서 캐릭터를 때리지 못하고 공격을 취소하는 일이 잦았다. 애니메이션 키 프레임 앞 부분의 8 프레임 정도를 삭제하고 재생 시간을 배속하여 공격이 매끄럽게 되게 수정했다.
  - 스테이지는 전부 유니티 기본 3D 오브젝트로 구현했다.

### 5. 기타

- 타이핑 및 단어 : 여러 적이 화면에 있을 때 어떤 단어든 먼저 입력하면 해당 적을 공격하는 직관적인 시스템을 구현했다. 각 단어의 오타인 개별 오타와 모든 입력 상태에 적용되는 전역 오타를 구분하여 억울한 오타 판정이 생기는 경험을 없앴다. 단순히 단어만 인식하면 되는 영어와 다르게 한국어는 초성, 중성, 종성을 달리 받아야하기에 내부적으로 초,중,종성을 분리하여 입력받는 시스템을 구현했다. 이는 나, 너 등의 복합 모음과 ㄱ, ㄴ 등의 복합 자음을 개별 자모로 분리하여 순차 입력 또한 지원한다. 단어 목록은 Oxford 5000와 국립국어원 표준국어대사전에서 구하였고 직접 작성한 Python 코드로 전처리하여 게임에 적합한 목록을 추렸다(특수 문자, 공백, 너무 긴 단어 제외).
- 공격 큐 : 단어를 성공적으로 입력하고 다음 공격이 발사되기 전에 또 다른 단어들을 입력 성공한 경우, 이후 공격이 무시되는 불쾌한 경험을 없애기 위해 공격 큐를 구현하여 모든 공격이 연속적으로 발사되게 구현했다.
- 로컬라이징 : 한국어, 영어, 스페인어, 프랑스어 지원으로 글로벌 사용자 대상의 게임을 구현했다.
- 레벨 디자인 : 쉬움 모드, 노말 모드, 하드 모드를 구분하여 게임이 쉽거나 어려운 유저들도 적절한 난이도로 즐길 수 있는 레벨 디자인을 구현했다.