2017년 졸업작품

팀 편돌이

**프로젝트 편돌이**

작성자 : 신창섭

변동사항

2016년 8월 16일 : 초안 작성. 플레이어 조작 관련 기획서 추가

2016년 8월 18일 : 플레이어 스탯 관련 기획서 작성 중

2016년 8월 22일 : 플레이어 설정 및 공식(18일자의 문서)에 대한 기획서 초안 완료

2016년 8월 25일 : 아이템 시스템 기획서 작성

2016년 9월 1일 : 아이템 중 장비에서 내구도에 대한 내용 추가.

Npc설정 및 상호작용 기획서 작성 시작

2016년 9월 4일 : 게임내 화면 레이어 구조 설정, 게임 내 천체의 구조

2016년 9월 9일 : 게임설계2 지도내용 반영 -> 시각화 및 간략화.

NPC 상호작용 기획서 및 퀘스트 기획서 작성

게임 내 천체에 대한 설정 수정(하늘 삭제, 클리퍼 내용 추가)

**목차**

[2. 게임의 기본 구조 5](#_Toc462768202)

[1. 게임의 기본 구조 5](#_Toc462768203)

[**2.2** **게임 내 천체의 표현** 5](#_Toc462768204)

[**2.2**  **게임 내 화면 레이어 계층** 6](#_Toc462768205)

[2. 플레이어 조작 및 상태 7](#_Toc462768207)

[2.1 **기획의도** 7](#_Toc462768208)

[2.2 개요 7](#_Toc462768209)

[**2.2** **대기** 7](#_Toc462768210)

[**2.3** **걷기** 8](#_Toc462768211)

[**2.4** **달리기** 8](#_Toc462768212)

[**2.5** **공격** 10](#_Toc462768213)

[**2.6** **회피** 11](#_Toc462768214)

[**2.8** **점프** 13](#_Toc462768215)

[**2.9** **아이템 사용** 14](#_Toc462768216)

[**2.10** **상호작용** 15](#_Toc462768217)

[3. 플레이어 설정 및 공식 16](#_Toc462768218)

[**3.1** **기획의도** 16](#_Toc462768219)

[**3.2** **개요** 16](#_Toc462768220)

[**3.3** **플레이어 테이블의 구성요소** 16](#_Toc462768221)

[**3.4** **각종 계산 공식(하단 모든 공식은 ‘전사’ 캐릭터를 기반으로 한다)** 18](#_Toc462768222)

[**3.5** **플레이어의 레벨업 및 레벨에 따른 플레이어 스탯의 상승** 21](#_Toc462768223)

[**3.6** **플레이어에 사용되는 애니메이션 종류** 22](#_Toc462768224)

[4. 플레이어 사망 시 23](#_Toc462768225)

[5. 아이템의 설정 24](#_Toc462768226)

[**5.1** **개요** 24](#_Toc462768227)

[**5.2** **Class에 대한 설정** 26](#_Toc462768228)

[**5.3** **내구도에 대한 설정** 27](#_Toc462768229)

[6. 상점시스템 27](#_Toc462768230)

[**6.1** **기획의도** 27](#_Toc462768231)

[**6.2** **내용** 28](#_Toc462768232)

[7. 퀘스트의 설정 29](#_Toc462768238)

[**2.2** **내용** 29](#_Toc462768239)

[**2.2** **개요** 29](#_Toc462768240)

[**2.2** **퀘스트 테이블의 구조** 29](#_Toc462768241)

[8. NPC 상호작용 31](#_Toc462768242)

[**2.2** **기획의도** 31](#_Toc462768243)

[**2.2** **개요** 31](#_Toc462768244)

[**2.2** **NPC라벨 설정** 33](#_Toc462768245)

[**2.2** **NPC와 상호작용 조건** 34](#_Toc462768246)

[**2.2** **NPC와 상호작용 시** 36](#_Toc462768247)

[**2.2** **기본 UI 호출** 37](#_Toc462768248)

[**2.2** **퀘스트** 38](#_Toc462768249)

[2.2 상점 이용이 가능한 NPC일 시에 40](#_Toc462768250)

[2.2 수리 기능이 가능한 NPC일 시 40](#_Toc462768251)

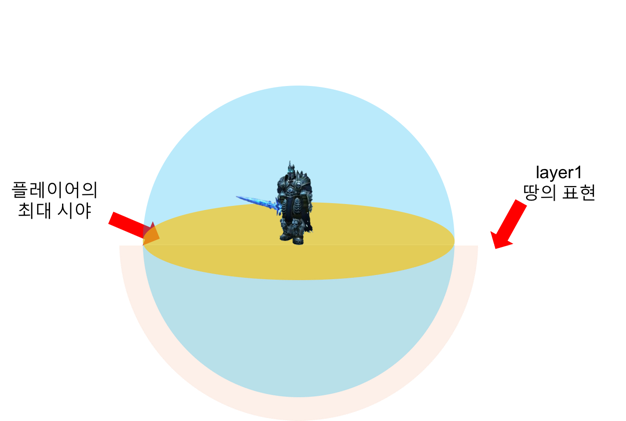
[9 드랍시스템 42](#_Toc462768260)

전투컨셉

## **게임 내 천체의 표현**

# **게임의 기본 구조**

## **게임 내 천체의 표현**



* + 1. 바닥 부분만 있는 반구 형태의 메시가 플레이어를 항상 따라다닌다.
    2. 항상 플레이어를 중심으로 절대좌표 기준 수평을 유지한다.
    3. **Layer1의 범위만큼 서버에서 데이터를 받아오고, 클라이언트에 표시한다.**
    4. 각각의 구는 플레이어의 움직임에 따라 회전하지 않고, 고정되어 있다.
    5. 각각의 지역에서 불러올 구의 이미지는 world 테이블에서 정의한다.

|  |  |
| --- | --- |
| No | 번호 |
| Name | 해당 월드의 이름 |
| Worldcode | 맵툴에서 정의할 해당 월드의 코드. 맵툴 상에서 어디서부터 어디까지가 해당 지역에 속하는지 정한다. |
| Sky1 | Layer1, 해당 월드의 배경 이미지 경로 |

## **게임 내 화면 레이어 계층**

* + 1. Layer 1
       1. 거리에 대한 값은 config.xml에서 <seedistance>의 값(m단위)로 정한다.
          1. <seedistance>에서 정해진 값에서 70%만큼 멀리 있는 물체는 텍스쳐를 제거하고 메시만 표시한다.
          2. <seedistance>만큼 멀어질 경우 해당 메시는 표현하지 않고 천체만 표현한다.
    2. Layer 2
       1. 맵툴로 생성한 기본적인 필드(바닥 부분)을 보여준다.
    3. Layer 3
       1. 플레이어들, 몬스터 및 각종 오브젝트를 보여준다.
    4. Layer 4
       1. HP바, 대화 말풍선, 이름 등의 스크립트를 보여준다.
    5. Layer 5
       1. 플레이어의 UI를 보여준다.



# **플레이어 조작 및 상태**

## **기획의도**

* + 1. 키보드에 최적화된 플레이어 조작방식 작성

## 개요

* + 1. 플레이어가 플레이를 할 때 기본적으로 사용하게되는 기본조작 / 상태는 **[대기/걷기/뛰기/공격/피해 입음/회피/점프/공격/아이템 사용/상호작용]**으로 나눌 수 있다. 각 상태에 따른 조작 방법 및 애니메이션 출력은 아래와 같다.

## **대기**

* + 1. 플레이어가 아무런 조작을 하고 있지 않을 경우 기본적으로 idle 애니메이션을 재생한다.
    2. 플레이어가 전투 모드에 돌입한 후에는 fightidle 애니메이션을 재생한다. 전투 모드로 변경되는 경우는 다음과 같다. fightidle 애니메이션 재생 이후 5초이상 추가로 fightidle 애니메이션을 재생하기 위한 조건을 충족시키지 못하면 다시 idle애니메이션을 재생한다.
       1. 직접 몬스터 / NPC / 사물 / 허공 등 모든 경우에서 공격행위를 할 경우
       2. 플레이어가 낙하 데미지 / 몬스터, 플레이어의 공격 등으로 피해를 입은 경우
       3. 플레이어가 피해를 입지 않더라도 플레이어의 위치를 강제로 이동시키는 모든 행위를 당할 경우

## **걷기**

* + 1. 기본적으로 키보드의 방향키를 사용해 플레이어를 조작할 수 있다.
    2. 이동 시에는 캐릭터가 진행방향을 바라보며 walk 애니메이션을 재생한다.
    3. 이동속도는 player 테이블의 moveSpeed값을 사용한다.
    4. 두 개의 키를 조합하여 대각선 이동이 가능하다.
    5. 동시에 3개 이상의 키를 누를 경우엔 1, 2번쨰 키를 제외한 키의 입력을 무시.



* + 1. 특정 방향으로 방향키를 누르다가 반대 방향의 키를 입력할 경우 입력을 무시
       1. 예를 들어 왼쪽을 누르고 있을 때 오른쪽을 누르면 입력을 무시한다.
    2. fightidle 돌입 이후, 플레이어가 이동조작을 하면 fightwalk 애니메이션을 재생
       1. Fightidle과 마찬가지로 5초 이상 fightwalk을 유지할 조건이 충족되지 않을 경우 walk 애니메이션을 재생한다.
       2. 예를 들어 만약 fightwalk 애니메이션 12프레임에 walk 애니메이션으로 전환될 조건을 충족했을 경우, walk 애니메이션의 12프레임부터 애니메이션이 시작된다.
    3. 방향키를 이용해 이동하다가 멈출 경우 가속도에 관계 없이 즉시 행동을 멈춘다.
    4. 이동조작이 불가능한 경우는 다음과 같다.
       1. 앉아있을 경우
          1. 앉은 상태에서 방향키를 입력할 경우, 캐릭터가 보고 있는 뱡항만 바뀐다.
       2. 점프 중일 경우
       3. 회피 동작 중인 경우
       4. 공격 / 스킬 사용 중인 경우
          1. 이 경우 예외로 공격 중 ‘방향’을 바꾸는 것은 허용된다.
       5. 상태이상 스킬에 걸려있는 경우(스턴 등)
       6. 강한 공격에 의해 경직된 경우
       7. 스킬 공격에 의해 넘어진 경우

## **달리기**

* + 1. 키보드의 A키를 누름과 동시에 이동 조작을 할 경우(방향키) 달릴 수 있다.
    2. 달리는 중에는 진행 방향을 바라보며 run 애니메이션을 호출
    3. 달리는 도중엔초당 기력(mp)를 plyaerSkill 테이블에 정의된 값만큼 소모한다.
       1. 예를 들어 만약 walk 애니메이션 12프레임에 run 애니메이션으로 전환될 조건을 충족했을 경우, run 애니메이션의 12프레임부터 애니메이션이 시작된다.
    4. 달리는 도중 A키를 손에서 뗄 경우, 달리기를 취소하며 진행 방향으로 walk 애니메이션을 재생한다.
       1. 1.2.3.1.1과 같이 같은 애니메이션 프레임에서 시작된다.

## **공격**

* + 1. 플레이어의 공격 수단은 [약한 공격 / 강한 공격 / 스킬공격]으로 구성된다.
       1. 각 공격에 따른 애니메이션이 종료되면 fightidle로 돌아간다.
    2. 약한 공격(S)를 누르면 플레이어가 보고 있는 방향으로 공격을 실행한다.
       1. 약한 공격에 대한 설정은 playerSkill 테이블에서 정의한다.
       2. 각 공격 애니메이션이 끝난 후 0.7초 이내에 같은 공격을 실행하면 다음 약한 공격으로 이어진다.
          1. 예를 들어 약한 공격1을 실행한 상태에서 연속으로 S버튼을 누를 경우 약한 공격2, 3, 4로 이어진다.
          2. 0.7초의 값을 confing.xml 테이블의 attackTiming에서 정의한다.
       3. 각 공격 이후 아무런 추가 조작을 하지 않을 경우 fightIdle 애니메이션으로 되돌아간다.
       4. 공격 도중 방향키를 입력할 경우, 다음 공격은 해당 방향으로 진행된다.
    3. 강한 공격(D)를 누르면 플레이어가 보고 있는 방향으로 공격을 실행한다.
       1. 강한 공격에 대한 설정은 playerSkill 테이블에서 정의한다.
       2. 각각의 강한 공격은 약한 공격 이후에만 실행이 가능하다.
          1. 약한 공격1 -> 강한 공격 1
          2. 약한 공격2 -> 강한 공격 2
          3. 약한 공격3 -> 강한 공격 3
          4. 약한 공격4 -> 강한 공격 4
       3. 각각의 강한 공격이 종료된 이후에는 fightIdle 애니메이션으로 되돌아간다.
    4. 각각의 스킬버튼(Q, W, E, R, T)을 누르면 플레이어가 보고 있는 방향으로 playerSkill 테이블에 정의된 각각의 스킬을 실행한다.
       1. 각각의 모든 스킬은 약한 공격, 강한 공격의 애니메이션이 진행중이라도 즉시 사용할 수 있다. 이 경우 해당 공격을 즉시 중지하고 스킬을 시전한다.
       2. 각각의 스킬 공격은 애니메이션이 끝나기 전까지 아무런 행동을 취할 수 없다.
    5. 공격 중 강한 공격 이상의 공격을 당했을 경우, 즉시 공격을 중지하고 피격 애니메이션을 실행한다. **이후 이와 관련된 설정은 플레이어 테이블 / 스킬 테이블에 power / resiPower 값을 설정해주는 것으로 정의한다.**

## **회피**

* + 1. 플레이어는 기본적으로 회피 스킬(A)을 사용해 적의 공격을 회피하거나, 순간적으로 먼 거리를 이동해야 할 때 등에 활용할 수 있다.
    2. 회피 스킬은 기본적으로 A키를 누르면 플레이어가 보고 있는 방향으로 사용되며, 이동 거리 / 소모 기력 등의 설정은 playerSkill 테이블에 정의된 값을 사용한다.
    3. 회피는 어떤 공격(약한 공격 / 강한 공격 / 스킬 공격) 중에도 해당 애니메이션을 취소하고 바로 사용할 수 있다.
    4. 회피 도중 모든 아이템을 사용할 수 없다.
    5. 플레이어가 회피 스킬 사용 도중 공격을 당했을 경우, 스킬 공격이 아니면 피해만 입히고 power 값에 따른 피격 애니메이션을 재생하지 않는다.
    6. 하지만 power 값에 따른 스킬공격 이상의 공격을 받으면 즉시 회피 애니메이션을 취소하고, 회피 완료 지점에 이동하면서 hardDam 애니메이션을 재생한다.
  1. **피해 입음**  
     플레이어가 피해를 입었을 때에 대한 설정이다.
     1. 플레이어의 피격 애니메이션은 받은 공격의 skill 테이블에 정의된 power 값에 따라 다르게 표출된다.
        1. 강한 공격에 피해를 당했을 때는 normalDam1, normalDam2 중 임의의 애니메이션이 표출된다.
           1. 단, 연속으로 강한 공격을 당할 경우 하나의 normalDam 애니메이션이 끝나기 전까지 다른 normalDam 애니메이션을 재생하지 않는다.
        2. 스킬 공격 이상의 강한 공격을 받을 경우 hardDam 애니메이션을 재생한다.

## **점프**

* + 1. 플레이어가 스페이스바(spacebar)를 누르면 제자리 혹은 최초 입력 방향으로 점프를 할 수 있다.
       1. 점프에 따른 체공 시간은 config.xml의 jump에서 정의하며, 정의된 시간(초)만큼 동안 0.5m만큼 상승하며 jump 애니메이션을 재생한다.
    2. 점프는 기본적으로 어떤 자원도 소모하지 않는다.
    3. 점프 도중 다른 방향으로 이동은 불가능하지만, 방향키를 누를 시 점프의 이동 방향은 유지 하면서 플레이어가 보고 있는 방향만 방향키의 방향으로 바뀐다.
    4. 점프 도중에는 playerSkill에 정의된 일부 스킬 이외에는 모든 행동이 불가능하다.
    5. 점프 도중에 공격을 당할 경우, 즉시 해당 공격 power에 맞는 애니메이션을 재생한다. 단, 원래 점프 착지 지점으로 이동하면서 애니메이션을 재생한다.
    6. 만약 점프를 할 때 플레이어의 위치에 천장 등이 있어 0.5m 만큼 점프를 하지 못할 경우, 애니메이션을 더 빠르게 재생해서 즉시 점프 동작을 마친다.
    7. 실제 공중 체공시간보다 애니메이션 시간을 약간 더 길게 설정하여, 바닥에 닿았을 때 착지 동작과 동시에 약간 미끄러지듯이 이동한다.



## **아이템 사용**

* + 1. 플레이어는 1 ~ 6 번호키를 사용해 등록된 아이템을 사용할 수 있다.
    2. 아이템은 플레이어가 idle, fightIdle 상태일 때만 사용 가능하다. Walk, run, fightIdle, , fightWalk, fightRun상태에서 아이템을 사용할 경우 즉시 자리에 멈춰서 아이템에 따른 애니메이션을 재생한다. 이후 idle / fightIdle 애니메이션으로 돌아간다.
    3. 아이템의 효과는 아이템에 따른 애니메이션이 완료 되었을 때 발휘된다. 아이템 사용 도중 회피를 사용해 아이템 사용을 취소했을 경우, 아이템은 소모되지 않는다.

## **상호작용**

* + 1. 플레이어는 상호작용이 가능한 대상이 근처에 있을 때 Spacebar를 누르면 해당 상호작용 대상을 바라보며 chat 애니메이션을 재생한다.

# **플레이어 설정 및 공식**

## **기획의도**

플레이어의 각종 수치에 대한 테이블화로 간편한 수정 및 추가가 가능하도록 한다.

같은 내용의 테이블을 서버에도 올려 서버에서 값을 내려 받도록 한다.

## **개요**

* + 1. 플레이어의 **[클래스 / 성별 / 스탯]** 등에 대해 테이블에서 정의할 수 있다.
    2. 추후 새로운 클래스가 필요하더라도 작업 소요를 적게 한다.
    3. 플레이어에게 보여질 값은 테이블의 값들을 사용해 별도 계산을 통해 노출된다.

## **플레이어 테이블의 구성요소**

* + 1. 플레이어의 테이블은 다음과 같이 성된다. **[번호 / 클래스 / 진영 / 캐릭터 이름 / 성별 / 무기타입 / 기본HP / 기본MP / 방어력 / 힘 / 민첩 / 지능 / 건강 / 이동속도 / 일반공격 스킯값 / 달리기 스킬 값 / 회피에서 스킬 값 / 스킬1 값 / 스킬2 값 / 스킬3 값 / 스킬4값 / 스킬5 값 / 플레이어 모델]** 이중 플레이어에게 노출되는 부분은 하단에서 다시 다룬다.
    2. 플레이어의 각 테이블 값은 stat 테이블에서 다음과 같이 정의된다.

|  |  |
| --- | --- |
| no | 캐릭터의 값(정수) |
| class | 캐릭터의 클래스 구분(정수)  0 : 기본 전사 클래스 |
| region | 캐릭터의 진영 구분  0 : 레지스탕스  1 : 얼라이언스 |
| name | 캐릭터의 기본 캐릭터 이름. 우리 게임은 졸작용임으로 그냥 완성형 한글 때려 박을 겁니다. |
| gender | 캐릭터의 성별을 결정합니다. 우리 게임은 기본적으로 하나의 모델링만 사용할 계획이므로 0만 사용하지만 일단 옵션은 만들어 둡니다.  0 : 남자  1 : 여자 |
| Hp | 캐릭터의 기본 체력(정수) |
| Mp | 캐릭터의 기본 기력(정수) |
| Def | 캐릭터의 기본 방어력(정수) |
| Str | 캐릭터의 기본 힘(정수) |
| Agi | 캐릭터의 기본 민첩(정수) |
| Int | 캐릭터의 기본 지능(정수) |
| Health | 캐릭터의 기본 건강(정수) |
| moveSpeed | 캐릭터의 기본 이동 속도(정수)  게임내 1m를 이동할 때 걸리는 시간을 기본으로 하며, 100을 입력하면 1m를 이동하는데 1초가 걸리는 것으로 계산 |
| normalAttack | 캐릭터가 사용하는 기본 약한공격1의 스킬번호. playerSkill 테이블의 No값을 입력한다. |
| runSkill | 캐릭터가 사용하는 기본 달리기 스킬의 스킬번호. playerSkill 테이블의 No 값을 입력한다. |
| dodgeSkill | 캐릭터가 사용하는 기본 회피 스킬의 스킬번호. playerSkill 테이블의 No 값을 입력한다. |
| skill1 | 캐릭터가 사용하는 스킬의 스킬번호. playerSkill 테이블의 no값을 입력한다. |
| skill2 | 캐릭터가 사용하는 스킬의 스킬번호. playerSkill 테이블의 no값을 입력한다. |
| skill3 | 캐릭터가 사용하는 스킬의 스킬번호. playerSkill 테이블의 no값을 입력한다. |
| skill4 | 캐릭터가 사용하는 스킬의 스킬번호. playerSkill 테이블의 no값을 입력한다. |
| skill5 | 캐릭터가 사용하는 스킬의 스킬번호. playerSkill 테이블의 no값을 입력한다. |
| model | 캐릭터의 모델링 파일 경로를 입력한다. |

## **각종 계산 공식(하단 모든 공식은 ‘전사’ 캐릭터를 기반으로 한다)**

* + 1. 크리티컬 발생 확률
       1. 크리티컬 발생 확률은agi 스탯과 아이템의 agi 값 보정을 받는다.
       2. 크리티컬의 확률은 그 최대 확률은 90%을 넘을 수 없다.
       3. 90%를 넘는 수치가 달성되면 그 이상의 확률을 차감해서 90%로 고정한다.
       4. 크리티컬 발생 시에는 최종 데미지가 200%로 증가되서 들어간다.
       5. 물리 크리 발생 확률과 마법 크리 발생 확률에서 공유해서 사용한다.

**기본 크리티컬 발생 확률 = SQRT ( SQRT ( 총 agi \* 500 ) ) \* 3**

* + - 1. 각 몹별로 def 수치에 따른 크리티컬 저항 수치가 존재한다.

**크리티컬 저항 확률 = SQRT ( SQRT ( 총 def \* 500 ) )**

* + - 1. 크리티컬 발생 확률은 다른 스킬 등에 의해 저하될 수 있다.
      2. 0% 이하로는 감소하지 않으며, 0% 아래로 내려갈 경우 0%로 고정시킨다.
      3. 최종 발생률은 [크리티컬 발생률 – 대상의 크리티컬 저항률]로 정한다.
      4. 유저에게 보여지는 크리티컬 값은 정수값만 사용한다.
      5. 0 이하의 값은 소숫점 한자리에서 반올림 한 값을 사용한다.
    1. 물리 공격력(다른 모든 것에 피해를 입힐 때)
       1. 플레이어의 물리 공격력은 str을 기반으로 한다.
       2. 공격력에는 최소 공격력과 최대 공격력이 존재한다.
       3. 만약 최소 공격력이 최대 공격력보다 높아질 경우, 최대 공격력은 최소 공격력의 수치로 고정된다.

**최소 공격력 = str \* 0.8**

**최대 공격력 = str \* 1.2**

* + 1. 마법 공격력(다른 모든 것에 피해를 입힐 때)

**마법 공격력 = int \* 1.5**

* + 1. 방어력(다른 모든 것에서 피해를 입을 때)

**방어력 = def**

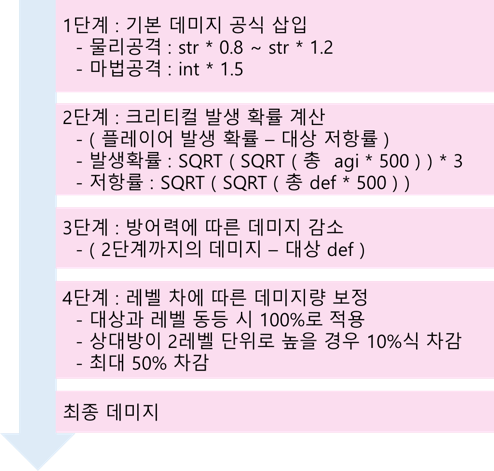
* + 1. 마법 방어력(다른 모든 것에서 피해를 입을 때)

**마법 방어력 = ( def \* 0.3 ) + ( health \* 0.7 )**

* + 1. 스탯에 따른 추가 HP 상승량
       1. 플레이어의 체력은 기본적으로 주어지는 HP에 str, health의 값을 종합해서 추가되게 된다 str, health 값에 따른 추가 HP의 값은 다음과 같다.

**HP = HP + (str \* 10) + health \* 10)**

* + 1. **최종 공격력 판별**
       1. 최종 공격량 판별은 아래와 같은 단계에 의해 계산된다.



* + - * 1. 기본 데미지 공식 대입
        2. 크리티컬 발생 확률 계산
        3. 방어력에 따른 데미지 감소
        4. 레벨 차에 따른 데미지량 보정

자기 렙과 동등하거나 낮은 경우 공격력이 그대로 들어가지만, 레벨이 올라갈수록 감소된 데미지가 들어가게 된다.

2레벨 단위로 최종 데미지가 10%씩 감소하며, 최대 50%까지 감소한다.

* + - * 1. 최종 데미지 산출
    1. HP 자동 회복
       1. 비전투 시(idle 상태일 시) 초당 자동 회복 체력 = (health / 2)
       2. 전투 중 초당 자동 회복 체력 = (health / 10)

## **플레이어의 레벨업 및 레벨에 따른 플레이어 스탯의 상승**

* + 1. 레벨업에 따른 스탯의 상승량은 playerLevelUp 테이블에서 정의한다. 각각의 값들은 기존 값에 더해지는 형태가 된다.
    2. 플레이어의 경험치 양은 기본적으로 0으로 시작된다. 그리고 현재 레벨 + 1의 요구 exp양을 달성하면 레벨업을 하며, 레벨업을 한 후에 남은 경험치는 다음 레벨의 경험치에 남는다.
    3. 레벨업 시에 플레이어의 체력이 모두 회복된다.

|  |  |
| --- | --- |
| No | 번호 |
| Class | 플레이어 클래스의 구분. 우리 게임은 졸작용이라 기본적으로 전사만 사용될 예정이지만, 일단 클래스 구분은 만들어둠(정수)  0 : 기본 전사  1 : 마법사 |
| Level | 해당 레벨 달성 시(정수) |
| Exp | 해당 level을 달성하기 위해 필요한 경험치의 양. |
| Def | 해당 레벨 달성 시 획득하는 스탯량 |
| Str | 해당 레벨 달성 시 획득하는 스탯량 |
| Agi | 해당 레벨 달성 시 획득하는 스탯량 |
| Int | 해당 레벨 달성 시 획득하는 스탯량 |
| Health | 해당 레벨 달성 시 획득하는 스탯량 |
| hp | 해당 레벨 달성 시 획득하는 스탯량 |

## **플레이어에 사용되는 애니메이션 종류**

|  |  |
| --- | --- |
| 애니메이션 | name |
| 대기 | normalIdle |
| 전투중 대기 | fightIdle |
| 전투중 걷기 | fightWalk |
| 걷기 | normalWalk |
| 전투중 달리기 | fightRun |
| 달리기 | normalRun |
| 대기중 딴짓 | IdleAction |
| 앉기 | Down |
| 약한공격1 | normalAtt1 |
| 약한공격2 | normalAtt2 |
| 약한공격3 | normalAtt3 |
| 약한공격4 | normalAtt4 |
| 강한공격1 | hardAtt1 |
| 강한공격2 | hardAtt2 |
| 강한공격3 | hardAtt3 |
| 강한공격4 | hardAtt4 |
| 스킬1 | Skill1 |
| 스킬2 | Skill2 |
| 스킬3 | Skill3 |
| 스킬4 | Skill4 |
| 스킬5 | Skill5 |
| 회피 | Dodge |
| 약한피해 | normalDam1 |
| 약한피해2 | normalDam2 |
| 강한피해 | hardDam |
| 사망 | Dead |
| 기절 | stun |
| 상호작용 | chat |

# 플레이어 사망 시

# 아이템의 설정

## **개요**

* + 1. 게임 내 사용되는 모든 아이템에 대한 통합 테이블을 만든다.

|  |  |
| --- | --- |
| no | 아이템의 번호 |
| name | 아이템의 이름(\n을 사용하여 줄나눔 가능) |
| outDescription | 아이템의 설명(\n을 사용하여 줄나눔 가능) |
| inDescription | 내부 공유용 아이템 설명. 유저에게 노출되지 않습니다. |
| class | 아이템 1차 분류(ex. 포션류, 무기류, 모자류). 이 부분에 따라서 아이템이 사용할 UI의 틀이 달라집니다. |
| subclass | 아이템 2차 분류(체력 정수 회복 포션, 마나 포션 등등) |
| quantity | 아이템의 갯수(예를 들어 골드 50개를 지급 한다던가.. |
| quality | 아이템의 등급(초록 푸른 등등의 등급 구분)  0 : 흰색(레거시)  1 : 초록색(노말)  2 : 파랑색(매직)  3 : 보라색(에픽)  4 : 금색(전설) |
| coolTime | 아이템의 사용 재사용 쿨타임. 정수 입력 |
| level | 아이템의 착용(사용) 가능 레벨. 해당 플레이어가 해당 레벨이 되어야 착용 / 사용이 가능합니다. |
| valueType | 아이템 판매시의 판매가치 타입. 기본적으로 골드 말고 캐쉬나, 특정 지역의 특산물 등으로도 거래가 가능할 수도 있으니까! 구입 시의 가격은 상점쪽 테이블에서 합니다.  0 : 골드  1 : 캐쉬 |
| value | 아이템 판매가치. valueType에 따른 상점에 되팔 때의 가격입니당. 이 값에 따라서 가격을 되돌려받습니다. |
| overLap | 아이템을 여러개 겹칠 수 있는가? 골드, 포션 등과 같은 소모품은 겹치게 할 수 있지만, 무기같은 장비류는 겹치지 않게 하기 위함입니다.  0 : 가능  1 : 불가능 |
| manNum | 최대 보유 갯수. 정수값을 입력합니다. 해당 아이템의 최대 보유 갯수를 설정합니다. 해당 갯수를 넘어가면, 새로운 슬롯에서 해당 아이템의 갯수가 1부터 시작됩니다. 단, 골드는 인벤토리에서 슬롯을 차지하는 형태가 아니므로 최대 허용 골드를 허용하면 나머지의 값들은 반환처리합니다. <- 이 부분은 에러처리 부분에서 추가로 다룰 예정입니다. |
| attSpeed | 플레이어의 공격 속도에 대한 설정. 실수로 입력할 수 있으며, 이 속도에 비례해 모든 공격 행위의 속도가 올라갑니다. 기본적으로 1초 걸리는 공격이었으면, 값이 100 증가함에 따라 0.1초 감소합니다. |
| critical | 플레이어의 공격 시 크리티컬 확률에 대한 설정. 정수로 입력할 수 있으며, %로 계산된다. 이 확률은 캐릭터의 크리티컬 확률을 올려주며, 절대값으로 추가 상승됩니다. |
| criDam | 플레이어의 크리티컬 공격 시 배율에 대한 설정. 기본적으로 상승하는 크리티컬 배수에 1당 1%씩 상승합니다. |
| def | 플레이어의 방어력이며 정수로 입력할 수 있다. |
| str | 플레이어의 힘이며 정수로 입력할 수 있다 |
| agi | 플레이의 민첩이며 정수로 입력할 수 있다 |
| int | 플레이의 지능이며 정수로 입력할 수 있다 |
| health | 플레이의 체력이며 정수로 입력할 수 있다 |
| hp | 플레이어의 체력에 대한 설정 |
| mp | 플레이어의 기력에 대한 설정. 정수로 입력할 수 있다. |
| moveSpeed | 플레이어의 이동속도에 대한 설정. 정수로 입력할 수 있으며 값 1에 따라 초당 이동 거리가 0.01m씩 증가합니다. |
| randomDrop | 랜덤박스일 경우의 드랍id입니다.(boxdropconfig.xml를 사용) |
| confirmDrop | 드랍 아이템이 확정 되어있는 박스일 경우 내장된 아이템의 번호 입력합니다. 여러개의 아이템이 들어있을 경우, 각 아이템은 |로 구분합니다(ex. 1232|5324|213) |
| iconImagePackage | 아이콘 아이콘이 들어있는 이미지 패키지의 번호 |
| iconName | 해당 패키지에서 아이콘 이름 |

## **Class에 대한 설정**

|  |  |  |
| --- | --- | --- |
| Class | subclass | 설명 |
| 0 | 0 | 재화-골드 |
| 0 | 1 | 재화-캐시 |
| 0 | 2 | 재화 – 경험치 |
| 1 | 0 | 한손장비 – 칼 |
| 1 | 1 | 한손장비 – 망치 |
| 2 | 0 | 양손장비 – 대검 |
| 2 | 1 | 양손장비 – 도끼 |
| 3 | 0 | 판금장비 – 모자 |
| 3 | 1 | 판금장비 – 어깨 |
| 3 | 2 | 판금장비 – 가슴 |
| 3 | 3 | 판금장비 – 허리 |
| 3 | 4 | 판금장비 – 다리 |
| 3 | 5 | 판금장비 – 다리 |
| 3 | 6 | 판금장비 – 발 |
| 3 | 7 | 판금장비 – 손목 |
| 3 | 8 | 판금장비 – 손 |
| 5 | 0 | 망토 |
| 5 | 1 | 목걸이 |
| 5 | 2 | 반지 |
| 5 | 3 | 보조장비 |
| 6 | 0 | 마법 아이템 – 스크롤류 |
| 7 | 0 | 소비 아이템 - 체력포션(정수값) |
| 7 | 1 | 소비 아이템 – 체력포션(%로 회복) |
| 7 | 2 | 소비 아이템 – 기력포션(정수값) |
| 7 | 3 | 소비 아이템 – 기력포션(%로 회복) |
| 7 | 4 | 소비 아이템 – 만능포션(체력, 기력 %로 회복) |

## **내구도에 대한 설정**

* + 1. 아이템의 클래스가 1, 2, 3일 경우 **테이블의 quantity는 장비의 내구도**로 작동하게 된다.
    2. 내구도가 0이 되면, 장비의 착용 효과가 발동되지 않는다.
    3. 장비의 내구도가 감소하는 경우
       1. 내구도가 존재하는 장비는 무기류 / 방어구류이다.
       2. 무기(class 1, 2)
          1. 플레이어가 특정 대상을 공격하는 경우

100회의 공격당 1씩 감소한다. 공격의 정의는 약한공격, 강한공격, 스킬공격을 모두 포함한다.

* + - * 1. 플레이어가 사망하는 경우

무기류의 내구도가 ( roundup ( ( 30 / 최대내구도 ) \* 100) 만큼 감소한다.

* + - 1. 방어구(class 3)
         1. 플레이어가 공격 당하는 경우

100회의 공격당 1씩 감소한다. 공격의 정의는 약한공격, 강한공격, 스킬공격을 모두 포함한다.

플레이어가 높은 곳에서 떨어져 낙하 데미지를 받았을 경우에는 4.3.3.3.1.1의 10배만큼 감소한다.

* + - * 1. 플레이어가 사망하는 경우

무기류의 내구도가 ( roundup(( 30 / 최대내구도) \* 100) 만큼 감소한다.

* + 1. 장비의 수리
       1. 장비의 수리는 수리 기능을 가진 npc에게 수행 가능하다. 수리비는 아이템의 등급, 플레이어의 레벨에 따라 다르게 책정된다.
          1. 흰색(레거시)

**소모된 내구도 \* (레벨 – 20) \* 0.02골드**

**\* 만약 (레벨 – 20)을 했을 때 값이 0 이하로 내려가면 1로 고정한다.**

**\* 장비 수리비의 최소 값은 1골드이다. 이하의 값은 1골드로 고정한다.**

* + - * 1. 초록색(노말)

**소모된 내구도 \* (레벨 – 20) \* 0.025골드**

* + - * 1. 파랑색(매직)

**소모된 내구도 \* (레벨 – 20) \* 0.03골드**

* + - * 1. 보라색(에픽)

**소모된 내구도 \* (레벨 – 20) \* 0.05골드**

* + - * 1. 금색(전설)

**소모된 내구도 \* (레벨 – 20) \* 0.1골드**

# 상점시스템

## **기획의도**

* + 1. 상점과 관련된 데이터를 통합적으로 관리하는 테이블을 만든다.

## **내용**

|  |  |
| --- | --- |
| No | 리스트 번호 |
| shopNo | 상점번호 |
| itemNo | itemList 테이블의 아이템 번호. |를 사용해 여러개를 입력할 수 있다. |
| levelLimit | 구입 레벨 제한 |
| moneyType | 구매할 수 있는 재화의 종류 |
| Money | 구매가격 |



# **퀘스트의 설정**

## **내용**

* + 1. 플레이어의 퀘스트에 사용될 통합 테이블을 만든다.

## **개요**

* + 1. 메인 / 서브 퀘스트의 값을 mainqurest.txt 테이블로 통합 관리한다.
    2. 동일한 테이블을 서버에도 올려 검증과정을 거친다.
    3. 일부 값들은 서버에서 내려준다.
    4. Npc와 상호작용하는 부분에 있어서 이후 퀘스트 추가 예상에 따른 더 많은 스키마를 필요로 하지만, 졸업작품에 그렇게 많은 기능을 넣지 못하므로 간략화.

## **퀘스트 테이블의 구조**

|  |  |
| --- | --- |
| No | 퀘스트 번호 |
| Type | 퀘스트의 타입을 정하는 값이다. 해당 타입을 정함에 따라 퀘스트의 조건을 설정한다.  0 : 대상과 대화하기  1 : 특정 몬스터를 잡기  2 : 특정 아이템을 모으기 |
| Para0 | 퀘스트 타입에 따른 파라미터 값 |
| Para1 | 퀘스트 타입에 따른 파라미터 값 |
| Para2 | 퀘스트 타입에 따른 파라미터 값 |
| Para3 | 퀘스트 타입에 따른 파라미터 값 |
| Para4 | 퀘스트 타입에 따른 파라미터 값 |
| startLv | 해당 퀘스트의 시작 가능 레벨 |
| preQuest | 해당 퀘스트를 하기 위한 선행 퀘스트 번호  해당 퀘스트를 클리어 하지 않았으면 퀘스트를 수행할 수 없다. |
| Reward1 | 퀘스트의 보상1 |
| Value1 | 퀘스트의 보상1의 갯수 설정 |
| Reward2 | 퀘스트의 보상2 |
| Value2 | 퀘스트의 보상2의 갯수 설정 |
| Reward3 | 퀘스트의 보상3 |
| Value3 | 퀘스트의 보상3의 갯수 설정 |
| Reward4 | 퀘스트의 보상4 |
| Value4 | 퀘스트의 보상4의 갯수 설정 |
| Reward5 | 퀘스트의 보상5 |
| Value5 | 퀘스트의 보상5의 갯수 설정 |
| qurestName | 퀘스트 이름 |
| questStartNpc | 퀘스트를 시작할 수 있는 npc. npcInfo 테이블의 값 |
| questEndNpc | 퀘스트를 완료할 수 있는 npc. npcInfo 테이블의 값 |
| questDeepHistory | 퀘스트의 상세 내용. questScript 번호 입력 |
| QuestHistory | 퀘스트의 간략 내용 |
| QuestCondition | 퀘스트의 조건 입력 |
| questStart | qeustScript에 쓸 퀘스트 스크립트 시작 번호 |
| QuestEnd | questScript에 쓸 퀘스트 스크립트 마감 번호. |
| questSuc | 퀘스트 완료 시 questscript 출력 스크립트 번호 |
| questBackground | 퀘스트를 부여한 NPC의 portrait. 0으로 입력할 경우 출력하지 않는다. |

* + 1. npcScript 테이블 연동

|  |  |
| --- | --- |
| No | No |
| FirstName | Npc의 성 |
| SecondName | Npc의 이름 |
| String | 출력 string |
| npcPortrait | Npc의 portrait |

* + 1. 선행퀘스트 미수행일 시
       1. preQuest를 수행하지 않았다면, 해당 퀘스트는 목록만 띄운다.
          1. 퀘스트 내용을 볼 수 없다.
    2. 너무 많은 퀘스트를 보유하고 있을 시
       1. 퀘스트를 포기 후 받으라는 메시지를 출력한다.
       2. **이후 UIstring에 적어 놓을 것!**

# NPC 상호작용

## **기획의도**

* + 1. NPC의 위치 및 정보를 통합 관리하는 테이블을 만든다. 이 값은 테이블에서 직접 입력할 수도 있고, 맵툴을 이용해 테이블에 입력할 수도 있다.
    2. 키보드 조작에 최적화 된 NPC 상호작용 UI를 만든다.

## **개요**

* + 1. 기본적인 위치 및 설정 입력 테이블
       1. npcInfo 테이블을 이용해 관리하게 된다.

|  |  |
| --- | --- |
| No | 넘버 |
| firstName | 해당 npc의 성 |
| lastName | 해당 npc의 이름 |
| Type | 해당 npc의 타입  0 인간형  1 사물형 |
| Chat | 대화 가능여부  0 불가능. 상호작용을 시도해도 아무런 반응이 없음  1 가능 |
| Region | 해당 npc의 진영 설정  0 중립(공격불가)작용  1 얼라이언스  2 레지스탕스  3 중립(공격가능) |
| basicChat | 상호작용을 시도했을 때 최초로 출력하는 대사. npcScript 테이블의 번호 입력 |
| Shop | 상점기능 개봉 여부  0 열리지 않음  1 열림 |
| shopBasicChat | 상점기능에 진입했을 경우 출력 대사. npcScript의 번호 입력 |
| shopBuy | 상점에서 물건을 구입했을 때의 출력 대사. npcScript의 번호 입력 |
| shopSell | 상점에서 물건을 판매했을 때의 출력 대사. npcScript의 번호 입력 |
| notMoney | 상점에서 물건을 구입 시도 하였으나, 금액이 부족할 떄의 출력 대사. npcScript의 번호 입력 |
| shopNo | Npc가 상점의 기능을 할 경우, 불러올 shop 테이블의 번호 |
| Repair | 수리기능의 개봉 여부  0 열리지 않음  1 열림 |
| completerepair | 수리 완료 시 출력 대사. npcScript의 번호 입력 |
| failRepair | 수리 실패 시 출력 대사. npcScript의 번호 입력 |
| World | 해당 npc가 위치하는 월드 명. |
| Px | 해당 npc의 px값 |
| Py | 해당 npc의 py값 |
| Pz | 해당 npc의 pz값 |
| Rx | 해당 npc의 rx값 |
| Ry | 해당 npc의 ry값 |
| rz | 해당 npc의 rz값 |
| model | 해당 npc의 모델링 파일 경로 및 파일명 |
| Portrait | 해당 npc의 초상화 |

* + - 1. npcScript 테이블 연동

|  |  |
| --- | --- |
| No | 번호 |
| Job | 해당 npc의 역할 |
| firstName | 해당 NPC의 성 |
| secondName | 해당 NPC의 이름 |
| String | 해당 스트링 |
| npcPortrait | 해당 초상화 |

## **NPC라벨 설정**

* + 1. NPC의 머리 상단에 npc의 이름 등을 표기한다.



* + 1. 진영에 따른 다른 색상 표기
       1. npcInfo의region 값이 0(중립, 공격 불가)
          1. 라벨 부분을 rgba(0, 255, 0, 100)으로 처리
       2. npcInfo의region 값이 1 혹은 2
          1. 아군 / 적군 여부에 따라 다르게 표출한다.
          2. 1의 진영에서 1을 볼 경우 혹은 2에서 2의 경우

라벨 부분을 rgba(0, 0, 255, 100)으로 처리

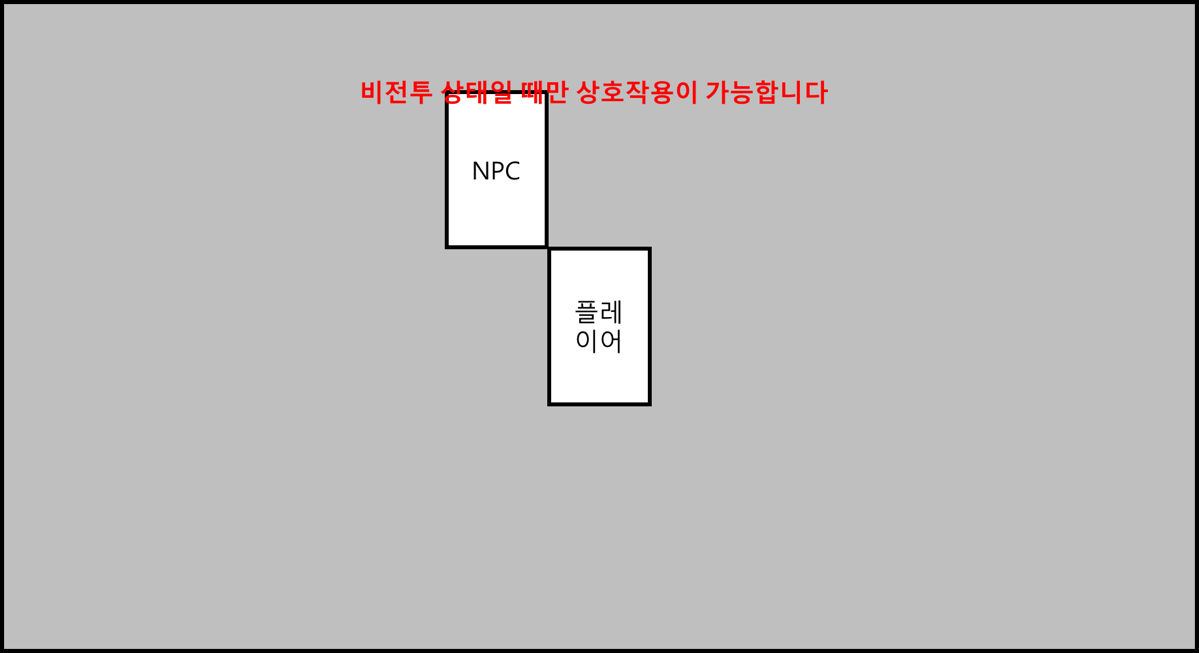
* + - * 1. 1에서 2, 혹은 2에서 1의 진영을 볼 경우

라벨 부분을 rgba(255, 0, 0, 100)으로 처리

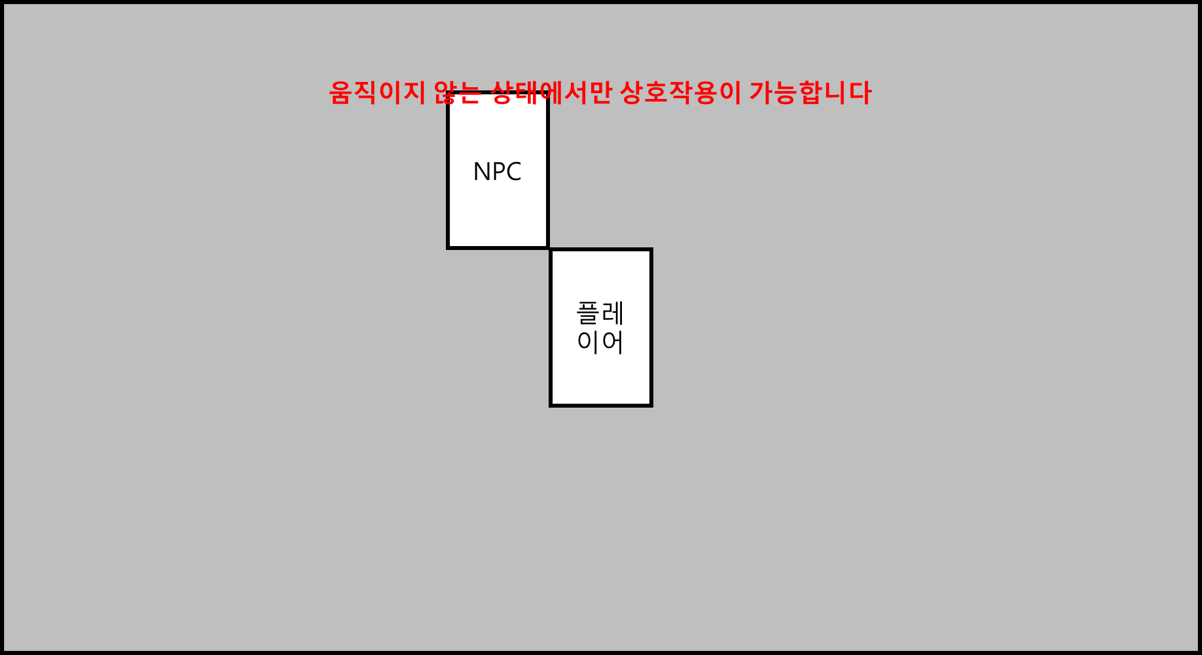
* + - 1. npcInfo의region 값이 3(중립, 공격 가능
         1. 라벨 부분을 rgba(255, 255, 0, 100)으로 처리

## **NPC와 상호작용 조건**

* + 1. 상호작용은 플레이어가 비전투 상태일 때만 가능하다.
       1. normalIdle, down 애니메이션 재생 상태에서만 가능하다.
       2. 이부분 추후 보강
          1. Chat 애니메이션을 호출한다.
       3. 전투 중 상호작용을 시도하면 에러 메시지 출력.(uiString)
          1. “비전투 상태일 때만 상호작용이 가능합니다”



* + - 1. 걷거나 달리는 등의 이동행위 중 상호작용을 시도 시 에러 메시지 출력(uiString)
         1. “움직이지 않는 상태에서만 상호작용이 가능합니다”

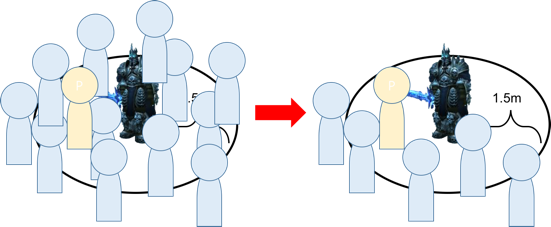


* + 1. 해당 npc의 일정 거리(1.5m)이내에 플레이어 캐릭터가 존재할 경우, 해당 npc의 머리 상단에 특정 이미지를 띄워준다.

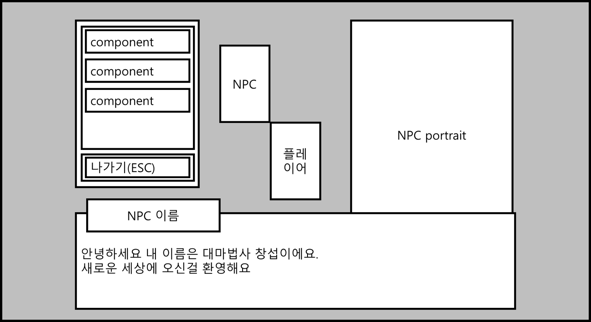


* + 1. npc가 여러명 존재할 경우 다음과 같은 우선순위에 따라 상호작용을 한다.
       1. 가장 가까운 NPC
       2. 플레이어가 정면으로 바라보고 있는 NPC
    2. 상호작용은 플레이어가 살아있을 때만 가능하다.
    3. 상호작용 대상 NPC가 인간형일 경우, 상호작용을 시도하면 Ry값이 돌아가며 플레이어를 바라본다. 이 기능은 클라이언트에서만 작동하여 본인에게만 보인다.

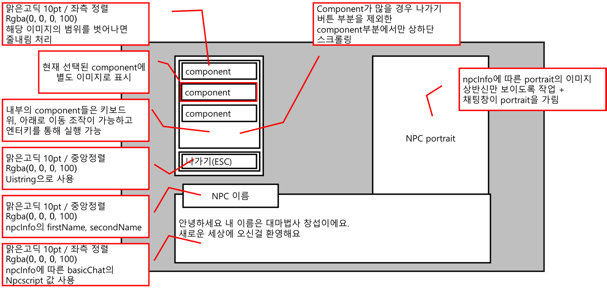
## **NPC와 상호작용 시**

* + 1.  NPC와 상호작용을 시도하면, 카메라가 줌 됨과 동시에 대상 npc 1.5m 이내의 다른 모든 플레이어는 일시적으로 보이지 않게 처리한다. 카메라가 줌 되는 값은 config.xml의 <npcChatCameraZoom>에서 정의한다.(단위 m)

## **기본 UI 호출**



* + 1. 내용 호출

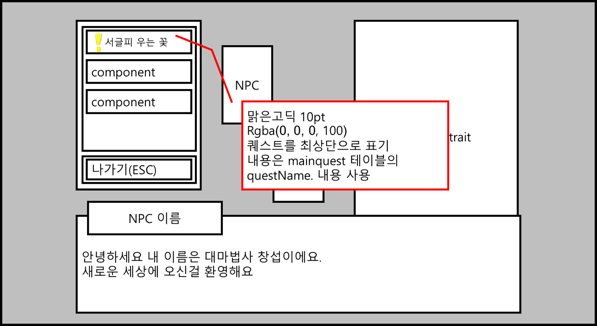


* + - * 1. ESC를 누를 시에 NPC와의 상호작용을 취소함
        2. componenet 최상단 노출 우선순위

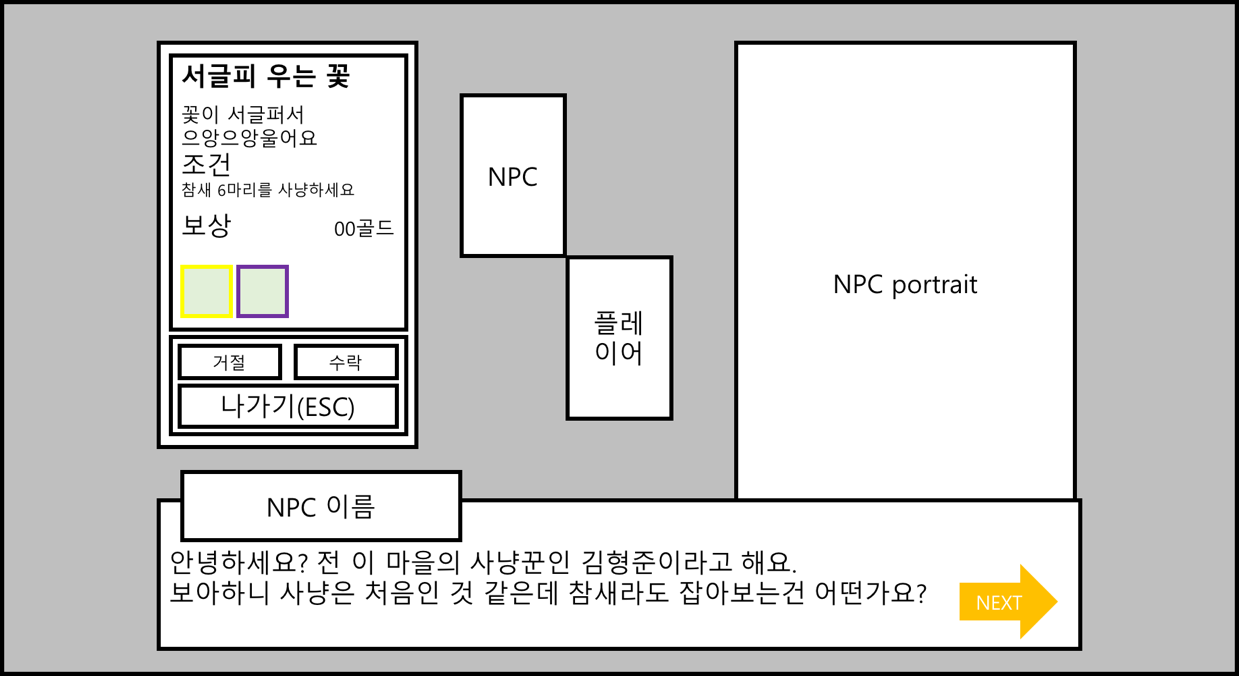
퀘스트 -> 상점 -> 장비수리 -> 그 외

## **퀘스트**

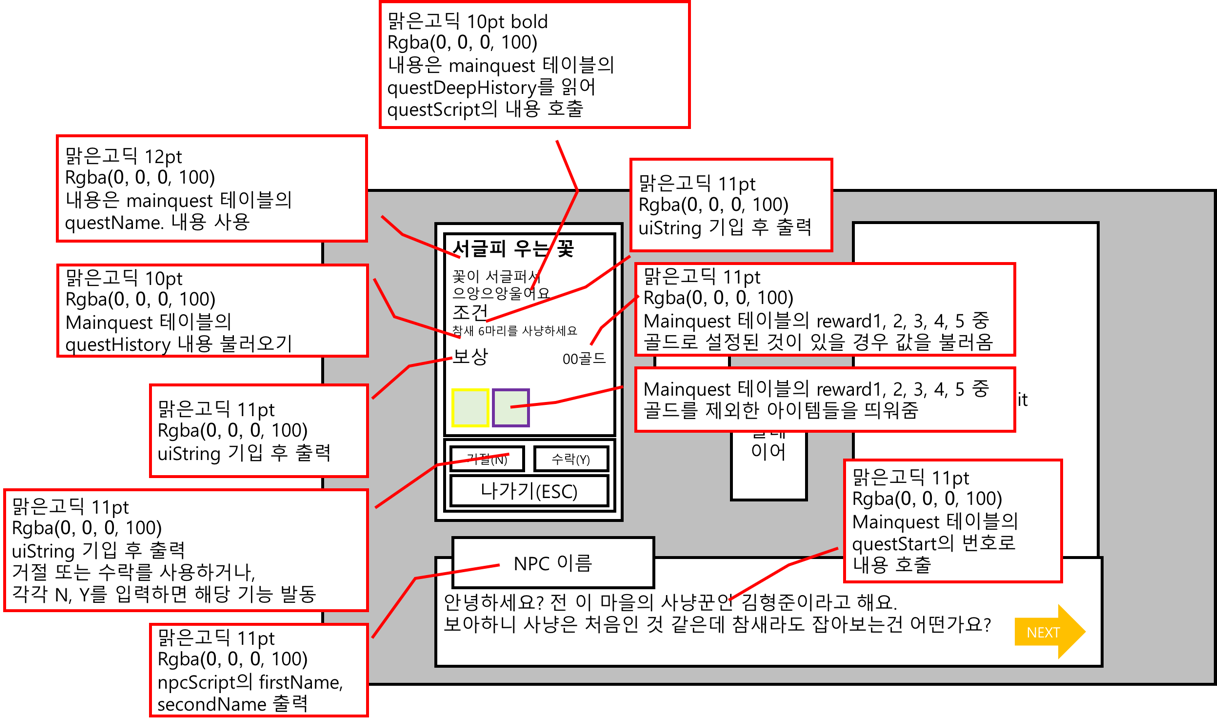
* + 1. 수행 가능한 퀘스트가 있을 시



* + 1. 퀘스트의 조건을 만족하지 않았을 시
       1. 따로 퀘스트를 표기해주지 않는다.
    2. 퀘스트의 시작 레벨보다 플레이어 레벨이 높을 경우
       1. 느낌표의 알파값을 50% 내려 표기해준다.
    3. 퀘스트의 수락
       1. 퀘스트를 진입하면 다음 과 같은 인터페이스를 띄워준다.

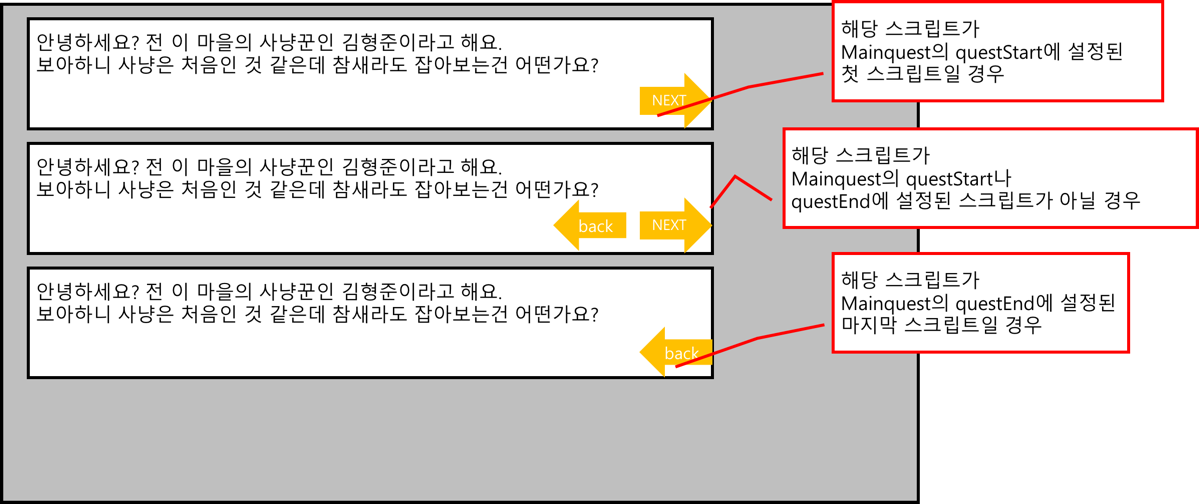


* + - * 1. 기본 퀘스트 수락 UI

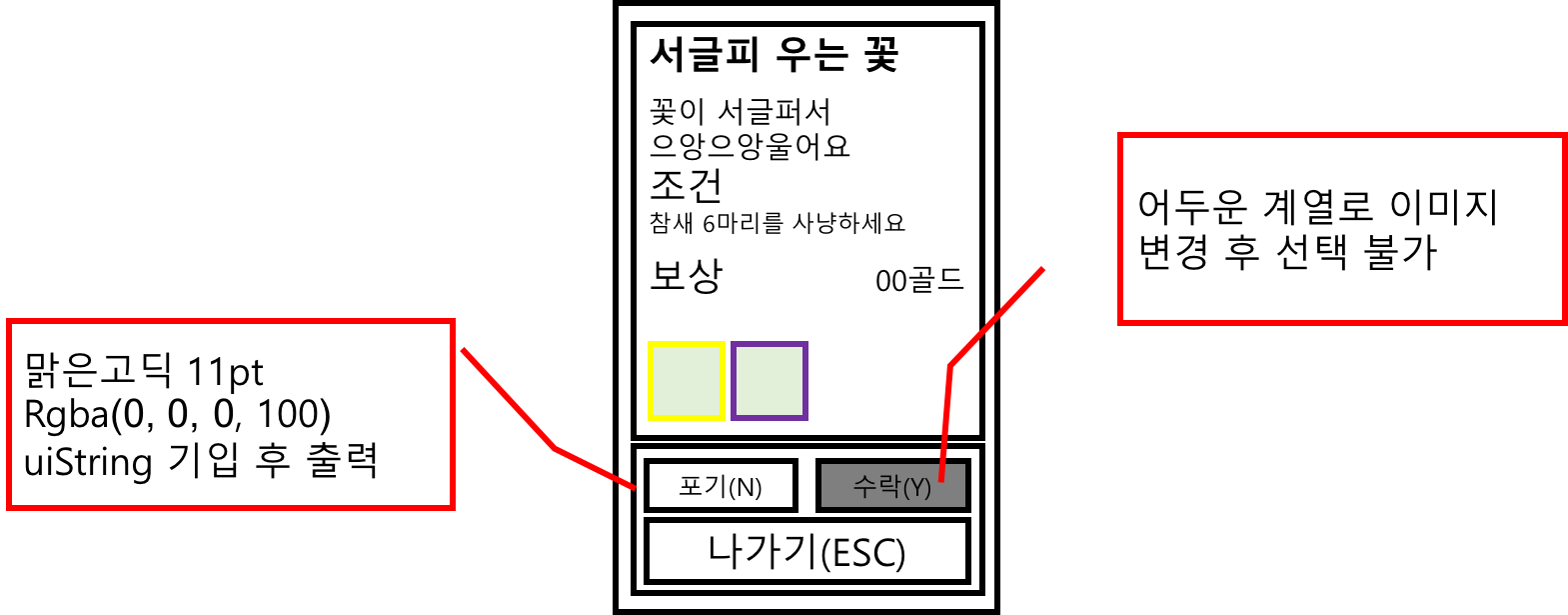


거절을 입력할 시에 기본 상호작용 UI 호출

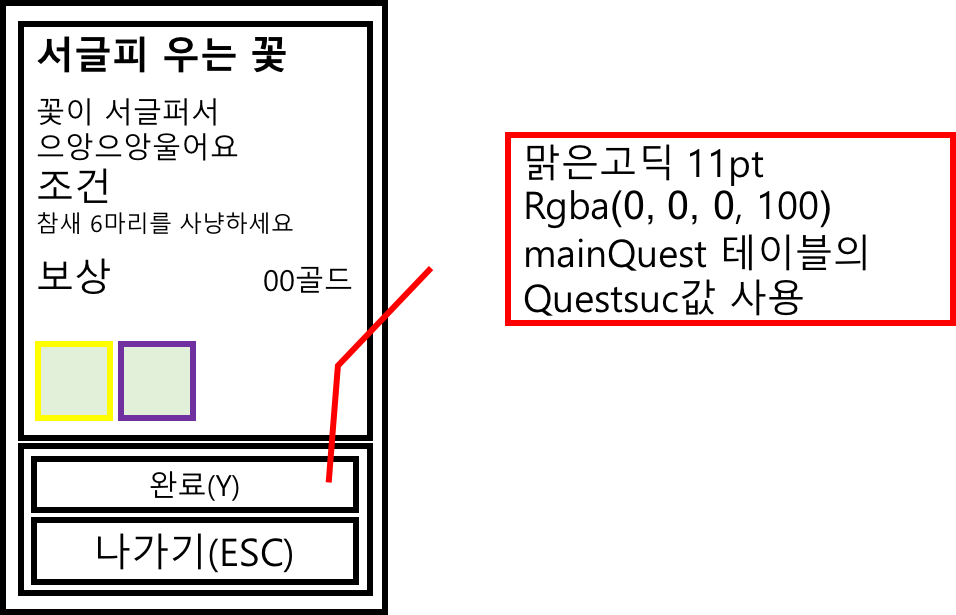
* + - * 1. npc 대화 세부 설정



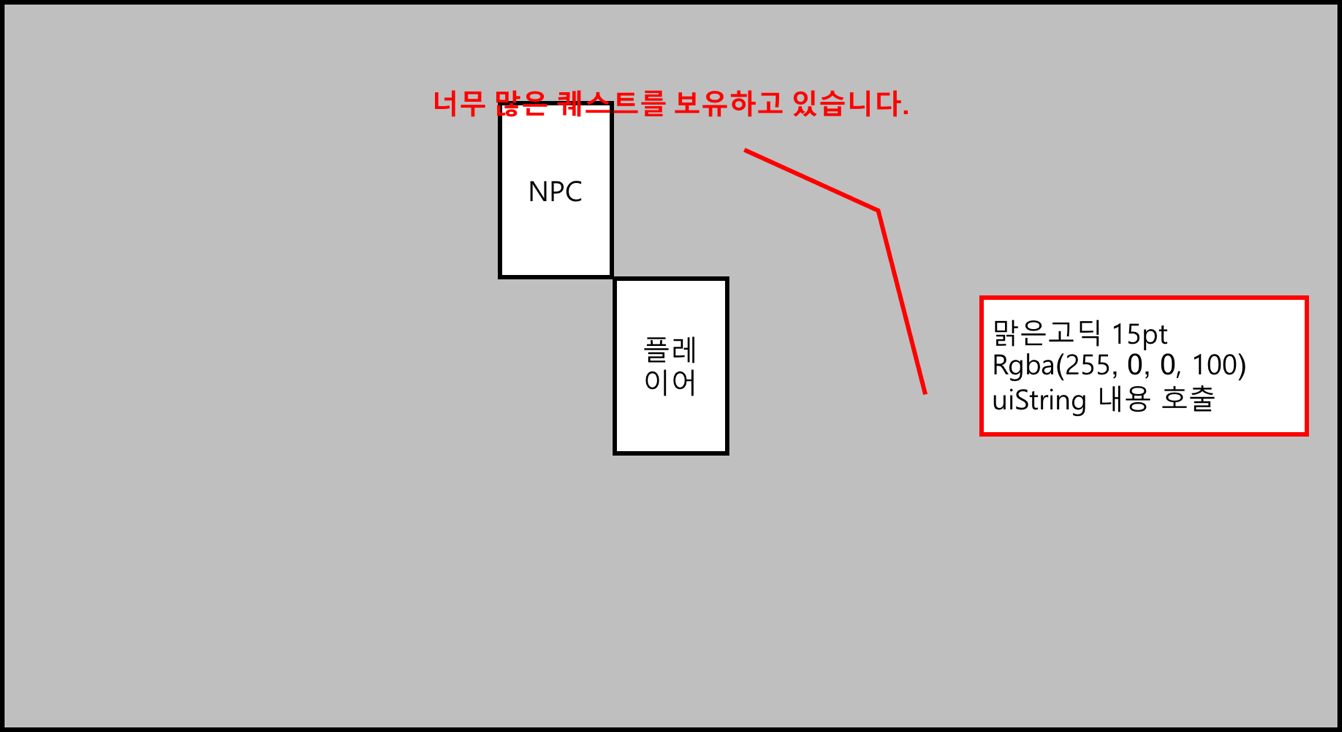
* + 1. 퀘스트를 수락 한 후



* + - 1. 하단 NPC 대화에는 questStart 테이블의 내용을 재호출 한다.
      2. 퀘스트 포기 시 관련된 보유중인 모든 퀘스트 아이템이 제거.
      3. 퀘스트 포기 시 모든 달성량 포기화(특정 몬스터 x회 잡기 등)
    1. 퀘스트 조건 만족 시



* + 1. 너무 많은 퀘스트를 보유하고 있을 시



## 상점 이용이 가능한 NPC일 시에

* + 1. 상점 기능이 들어간다고 수정 시 기입

## 수리 기능이 가능한 NPC일 시

* + 1. 수리 기능이 들어갈 시 이후 수정



# 드랍시스템