# GRID SAGA기술 문서

# 개요

## 1.1 작성 목적

본 문서는2025/08/08부터 진행되는 Grid Saga의 개발을 위한 기술 설계 지침 문서이다.

## 1.2 프로젝트 요약

Grid Saga의 명칭은 유니티 2D Grid상에서 이루어지는 서사시, 무용담이라는 의미에서 Grid에 Saga를 합성하여 Grid Saga로 명명하였다. 해당 게임의 컨셉은 실시간 턴제 방식으로 두 진영이 한 필드에서 전투를 치르는 방식이며, 전투 타이머를 기반으로 전투 타이머 내에서 상대 진영의 hp를 모두 제거한 쪽이 우승하는 방식을 따른다. 타이머를 넘기거나 아군 진영 hp가 모두 제거될 경우 패배 처리한다.

# 개발 환경

|  |  |
| --- | --- |
| 개발 환경 | 용도 |
| Django (Python) | 백엔드 서버 |
| SQlite | DBMS |
| Unity (C#) | 클라이언트 엔진 |

# 시스템 아키텍처

텍스트, 스크린샷, 원, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

Grid Saga 시스템 아키텍처 구조는 Django 공식 MVT 구조를 따르되, T. MVT 구조는 Model – View – Template형식의 시스템 아키텍처를 말하며 Template는 클라이언트를 담당하며, View는 클라이언트 요청에 따른 로직을 수행하고 이를 반환한다. Model은 DB 스키마 정의 및 ORM을 통한 DB 관리를 담당한다.

본 프로젝트의 시스템 아키텍처 구조는 해당 구조에 따라 Template은 Unity, View와 model은 Django로 한다. 이로써 렌더링, 게임 진행 코드는 유니티에 두고 데이터 송수신 및 관리는 Django에서 수행하도록 분리한다. Unity – Django 간 통신은 REST를 이용한다.

# 게임 설계

## 4.1. 게임 흐름 설계

### 게임 플레이 용어 정의

* 플레이어 파티: 4캐릭터를 1 파티로 규정한다
* 적: 적의 수는 1 ~ 4 개체를 1 파티로 규정한다
* 스테이지: 일반 몬스터 또는 보스가 발생하는 한 번의 전투가 벌어지는 공간
* 계층: 첫 일반 몬스터 ~ 보스까지 스테이지 묶음을 한 계층으로 정의한다
* 계층 타이머: 1계층 당 존재하는 시간 제한 타이머
* 피로도: 던전 진행에는 계정 별로 일일 횟수 제한을 둔다, 해당 제한 자원은 피로도로 명명

### 4.1.2. 게임 진행 방식

게임 진행은 pve 던전 공략 방식을 따른다, 다만 pvp 확장 가능성을 염두하고 설계한다.

* 던전입장: 플레이어 선택에 의해 던전에 입장, 피로도를 차감하여 DB에 반영한다
* 스테이지 진행: 스테이지를 순서대로 진행하며 계층 타이머가 모두 소진되기 전에 스테이지의 적을 전멸시키면 다음 스테이지로 이동한다
* 계층 클리어: 계층의 보스를 쓰러뜨리면 다음 계층으로 진행이 가능하다
* 타이머 규칙: 1 계층 1 계층 타이머 방식을 따른다, 계층에 플레이어가 진입하면 작동을 시작하고, 다음 계층으로 이동하면 해당 계층의 타이머를 새로 시작한다, 타이머가 다 되면 나가기 처리한다
* 나가기 조건: 파티 전멸, 타이머 아웃, 유저 의지에 따른 던전 나가기 처리
* 플레이어 선택: 던전 진행 중 방식을 수동 입력과 자동 진행으로 선택 가능

### 던전 설계 방식

* 계층: 던전은 첫 몬스터 ~ 보스 몬스터의 묶음인 계층으로 나뉘어진다, 계층을 클리어하면 보상을 지급한 후 다음 계층으로 진행할지 선택 가능하게 한다
* 보상: 플레이어가 보스 스테이지를 넘겨 계층을 클리어하면 아이템 제공
* 난이도: 플레이어가 계층을 클리어하면 다음 계층 몬스터를 변경 및 강화를 병행하여 레벨 스케일링, 스탯 전반에 대한 퍼센트에 따른 레벨 스케일링 예정이나 정확한 퍼센트 수치 값은 미정 상태
* 캐릭터 사망: 계층 진행 중 전원이 사망하면 나가기, 한 캐릭터가 남아서라도 계층을 클리어 하면 생존 캐릭터는 100% hp로 회복, 사망한 파티 캐릭터는 hp를 50%로 부활
* 추가보상: 플레이어가 계층 진행 중 자동 진행을 한 번도 사용하지 않고 계층을 진행하면 추가 보상 제공, 이 외에는 특정 제한 시간 이하에 계층 클리어(ex: 타이머가 10분인 계층을 3분 내 클리어) 또는 파티 인원 전원 생존 상태에서 계층 클리어 등 고려
* UI 고려 사항: 현재 몇 계층인지, 남은 계층 타이머가 몇인지, 캐릭터 스킬 사용 가능 여부, 오토의 경우 스킬 사용 순서 큐, 파티 총합hp와 상대의 총합 hp

## 4.2. 캐릭터 설계

## 4.3. 아이템 설계

아이템은 캐릭터 강화 아이템과 장착 아이템으로 나뉜다.

### 4.3.1. 캐릭터 강화 아이템

캐릭터 강화 아이템은 캐릭터에게 exp를 제공하는 1회성 재화이다. ItemType은 Consume으로 정의한다.

* 효과: 특정 캐릭터에게 소비할 경우 exp를 증가, exp가 다 차면 캐릭터 레벨이 증가하여 스탯 증가

### 4.3.2. 장착 아이템

장착 아이템은 캐릭터에게 장착되는 아이템으로 영구적이다. ItemType은 Equip으로 정의한다.

* 효과: 장착 아이템을 소지한 캐릭터가 가진 스테이터스 중 하나를 퍼센트 또는 값만큼 증가
* 아이템 스킬: 특정 액티브 스킬을 캐릭터에게 적용
* 제한: 캐릭터가 아닌 장비가 진행 주체가 되는 현상을 막기 위해 패시브 스킬을 아이템 스킬에 배치는 금함

## 4.4. 스킬 설계

스킬은 패시브 스킬, 액티브 스킬로 나뉜다. 두 스킬은은 모두 트리거를 기반으로 작동하나 트리거의 조건에서 차이가 발생한다.

### 4.4.1. 패시브 스킬

패시브 스킬은 트리거를 통해 자동 발동되는 스킬이다. 주로 캐릭터 자신에 대한 버프를 효과로 한다.

* 트리거: 캐릭터 체력 상황, 던전 환경, 조우한 적 타입
* 효과: 패시브 스킬을 소유한 캐릭터의 스테이터스를 증가

### 4.4.2. 액티브 스킬

액티브 스킬은 수동 모드에서 플레이어의 직접 입력, 자동 모드에서 큐 스택 차례를 트리거로 받아 발동되는 스킬이다. 한 번 사용 후 다음 사용까지 쿨다운이 필요하며 상대에게 직접 피해, 디버프, 아군에게 일시적인 강한 버프 등의 영향을 끼치는 스킬을 구조로 한다.

* 트리거: 플레이어의 직접 입력, 자동 모드에서 캐릭터의 큐 스택 차례가 올 경우
* 효과: 상대 및 아군에게 직접 피해, 일시적인 강한 버프, 디버프 영향
* 조건: 강한 성능을 가지지만 쿨다운이 필요

# 5. UX/UI

## 5.1. 로그인 UI

로그인 UI는 플레이어가 게임 실행 직후 마주치는 UI로서 유저의 로그인 가능 여부를 다룬다

* 인증: 아이디, 패스워드를 입력 받아 Authenticate하기 위한 창
* 새 계정 생성: 새로운 유저가 새 계정을 생성하기 위한 창

## 5.1. 로비 UI

## 5.1. 던전 UI

# 구현 세부 사항

# 프로젝트 목표