

Национальный исследовательский ядерный университет «МИФИ»
Институт интеллектуальных кибернетических систем
Кафедра №42 «Криптология и кибербезопасность»



Отчёт

**О выполнении лабораторной работы №1
«Построение модели данных»
по курсу «Безопасность баз данных»**

Студент: Лузганов К. А.
Группа: Б22-525

Москва – 2024

Предметная область

Предметная область представляет из себя сервис, торгующий внутриигровым снаряжением для некоторой RPG в виде уникальных токенов. В данном случае – оружием. Для представления игрокам актуальной информации об оружии сервису нужно иметь данные о характеристиках и стоимости оружия, о том, какие игроки каким оружием владеют, и о том, какие монстры имеют уязвимости (или сопротивления) к урону конкретного оружия – чтобы предлагать приключенцам эффективное решение их проблем. Листинг инструкций SQL, использованных для создания базы данных, приведён в приложении А.

Спецификация таблиц

Таблица «Weapon» (см. рис. 1) содержит основную информацию об оружии, идентифицируя его по названию.

- name – уникальное название оружия;
- basic_damage – значение урона без модификаторов;
- damage_type – тип наносимого урона (дробящий, огненный, некротический и т.д.);
- rarity – редкость оружия (обычный, эпический, легендарный и т.п.);
- price – цена оружия во внутриигровых единицах;
- description – подробное атмосферное описание оружия.

Таблица «Player» содержит информацию об именах игроков, которые определяются по их уникальным идентификаторам.

- id – уникальный идентификатор игрока;
- fraction – название фракции, к которой принадлежит игрок (каждый игрок принадлежит не более, чем к одной фракции);
- name – имя игрока.

Таблица «Entity» является промежуточной для установления отношения «многие ко многим» между игроками и оружием.

- id – уникальный идентификатор объекта оружия (не путать с более общим классом оружия, который соответствует таблице Weapon);
- owner_id – уникальный идентификатор игрока, владеющего оружием;
- weapon_name – название класса оружия, реализацией которого является данный объект;
- weapon_level – текущий уровень конкретного оружия.

Таблица «Weaknesses» содержит столбцы:

- Creature – название существа;
- damage_type – тип урона (колющий, дробящий, огненный, некротический и т.д.);
- damage_modifier – величина (в процентах), на которую будет домножен базовый урон оружия при получении этого типа урона соответствующим существом.

Таблица «Bestiary» содержит информацию о различных существах, их опасности и месте обитания, чтобы игрок мог, зная своего врага, подобрать оружие с соответствующим типом урона или наоборот – подбирая оружие, сразу видеть, против кого оно будет эффективно.

- creature – уникальное название существа;
- biome – место обитания;
- damage – показатель опасности существа.

Таблица «Fraction» определяет, каким оружием может пользоваться персонаж конкретной фракции. Одна фракция предоставляет один класс оружия. Данная таблица

также ставит в соответствие каждой фракции не более одного существа, которое является заклятым врагом этой фракции.

- fraction_name – ключевое поле, уникальное название фракции;
- weapon_name –класс оружия, к которому даёт бонус данная фракция;
- enemy_name – название существа – избранного врага фракции (см. таблицу «Bestiary»).

Таблицы соответствуют третьей нормальной форме:

- на пересечении строки и столбца находятся атомарные значения;
- в каждой таблице присутствует ключ;
- в таблицах «Ownership», «Improvement», «Weaknesses» каждое поле существенно зависит от каждой части составного ключа; в других таблицах ключ не является составным;
- ни один неключевой столбец не определяется другими неключевыми столбцами.

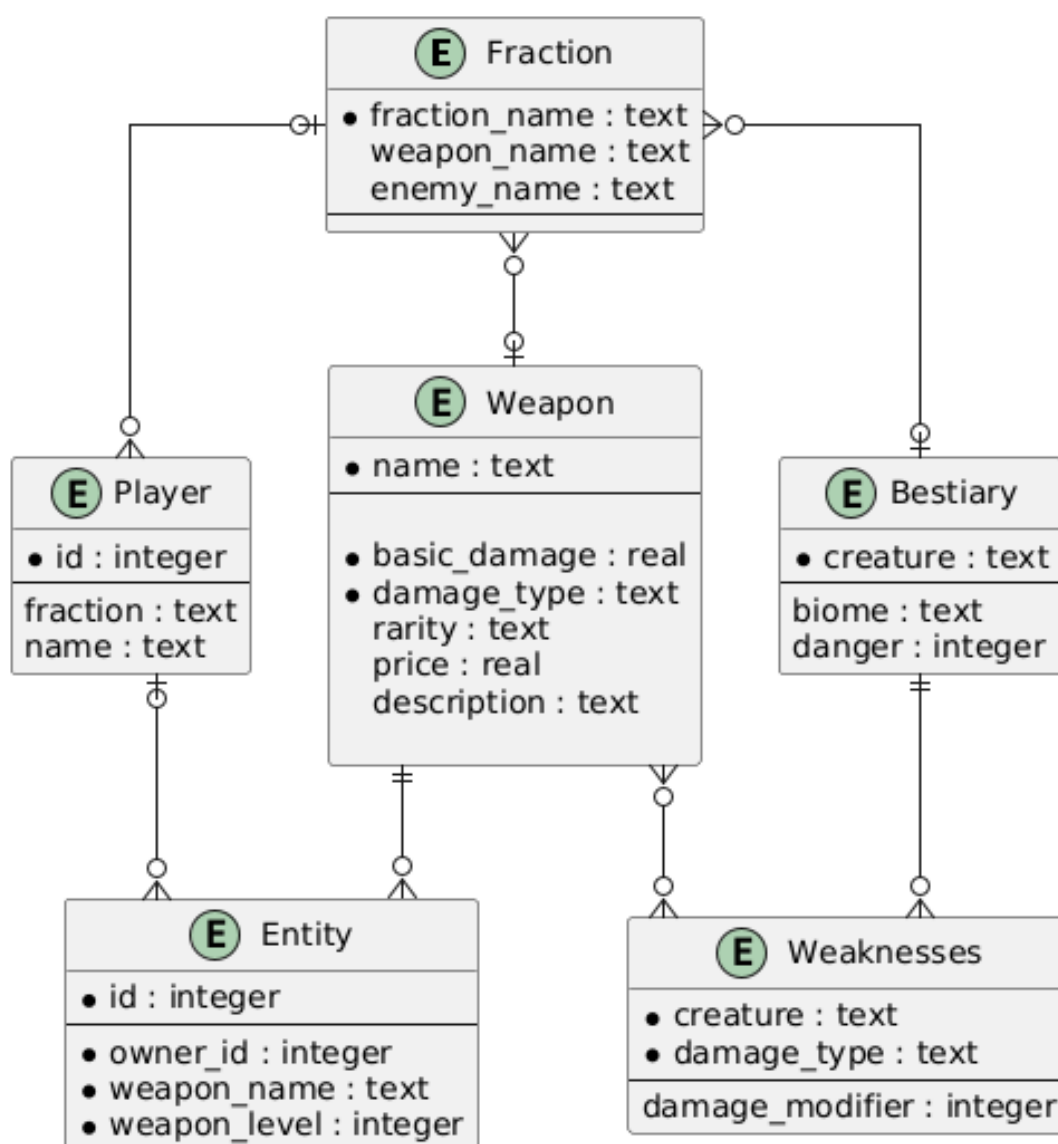


Рисунок 1. – Диаграмма отношений сущностей

Заключение

Была разработана структура базы данных, освоен инструмент построения диаграмм отношения сущностей *PlantUML* и правила построения этих диаграмм (в нотации «воронья лапка»). Были изучены средства создания таблиц с помощью инструмента *Sqlite3* и создана база данных, соответствующая разработанной структуре. Листинг инструкций SQL, использованных для создания базы данных, приведён в приложении А.

Приложение А. Листинг использованных инструкций SQL

```
PRAGMA foreign_keys = ON;

-- Таблица оружия
CREATE TABLE Weapon (
    name TEXT PRIMARY KEY,
    basic_damage REAL NOT NULL,
    damage_type TEXT NOT NULL,
    rarity TEXT,
    price REAL,
    description TEXT
);

-- Таблица игроков
CREATE TABLE Player (
    id INTEGER PRIMARY KEY,
    faction TEXT,
    name TEXT
);

-- Таблица сущностей (владение оружием)
CREATE TABLE Entity (
    id INTEGER PRIMARY KEY,
    owner_id INTEGER NOT NULL,
    weapon_name TEXT NOT NULL,
    weapon_level INTEGER NOT NULL,
    FOREIGN KEY (owner_id) REFERENCES Player(id) ON DELETE CASCADE,
    FOREIGN KEY (weapon_name) REFERENCES Weapon(name) ON DELETE CASCADE
);

-- Таблица слабостей существ
CREATE TABLE Weaknesses (
    creature TEXT NOT NULL,
    damage_type TEXT NOT NULL,
    damage_modifier INTEGER NOT NULL,
    PRIMARY KEY (creature, damage_type)
    FOREIGN KEY (creature) REFERENCES Bestiary(creature) ON DELETE CASCADE
);

-- Таблица бестиария
CREATE TABLE Bestiary (
    creature TEXT PRIMARY KEY,
    biome TEXT NOT NULL,
    danger INTEGER NOT NULL
);

-- Таблица фракций
CREATE TABLE Fraction (
    faction_name TEXT PRIMARY KEY,
    weapon_name TEXT,
    enemy_name TEXT,
    FOREIGN KEY (weapon_name) REFERENCES Weapon (name) ON DELETE CASCADE
    FOREIGN KEY (enemy_name) REFERENCES Bestiary(creature) ON DELETE CASCADE
);
```