# 卡牌游戏软件

# 需求分析与概要设计

## 项目说明

## 项目目标：

开发一款基于Unity引擎的叙事驱动型卡牌对战游戏包含以下特性：

核心玩法​​：回合制卡牌对战，融合策略性卡牌组合、资源管理与剧情推进

叙事特色​​：通过卡牌对话、场景交互推进黑暗童话风格的主线剧情

扩展性​​：模块化卡牌系统支持后续扩展（如新卡牌类型、特殊规则）

## 软硬件环境需求

开发环境 Unity 2022 LTS、C#、Visual Studio 2022

运行平台 Windows 10/11

存储空间 本地SQLite数据库/csv文件格式（卡牌数据） + PlayerPrefs（进度存档）

## 使用的关键技术：

状态机设计​​：基于有限状态机（FSM）控制战斗流程（抽牌→出牌→结算）

数据驱动​​：ScriptableObject实现卡牌属性

技术难点​​：

卡牌组合效果动态计算

跨场景数据持久化

## 需求分析

## 系统用例

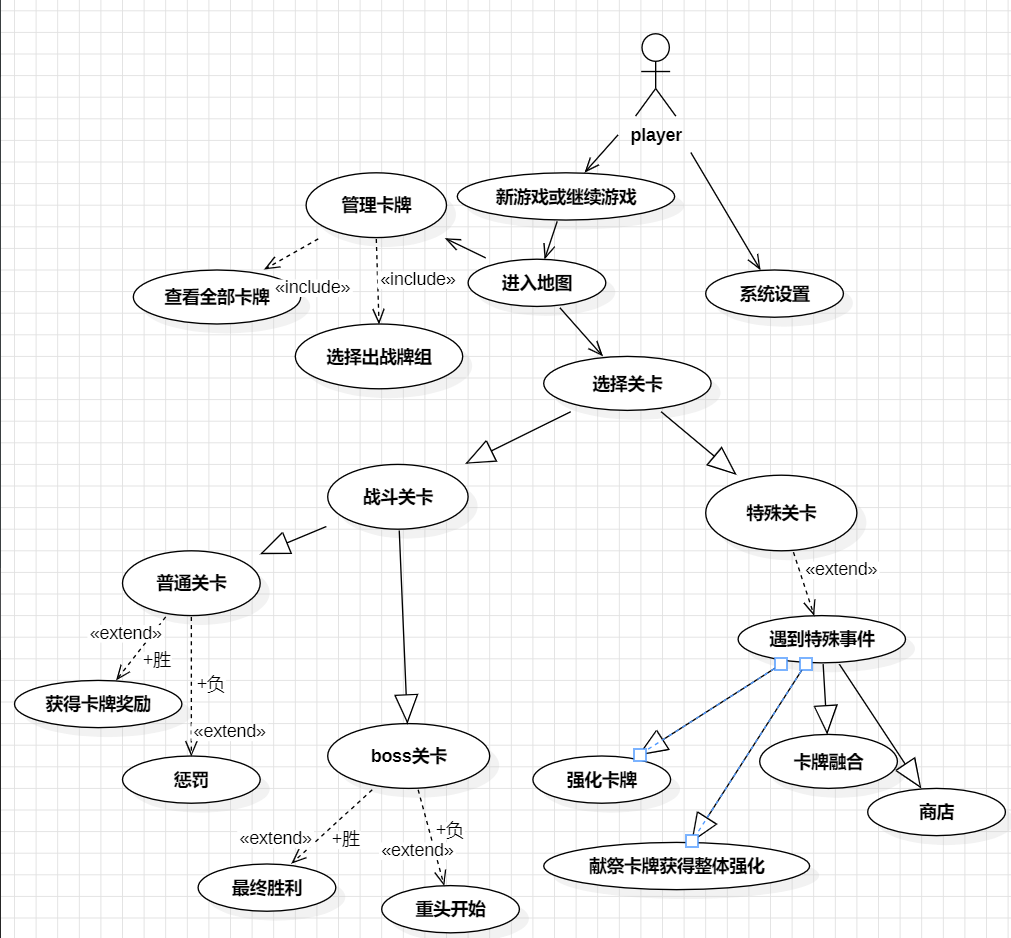


图 1 系统用例图

**参与者**：玩家

**用例图核心节点**：

1. 卡牌对战（基础用例）

- 扩展用例：特殊规则触发（如献祭机制）

2. 剧情推进（包含分支选择）

3. 卡牌收藏管理（包含卡牌升级/分解/融合/献祭）

4. 游戏设置（音效/画质调节）

……

说明：

（1）用例图使用StarUML或华为的CodeArts Modeling绘制，要画出用户角色和对应用例

（2）用例是描述用户使用软件的场景，不要把用户看不到的后台功能画在用例图上。这里的用户不仅是人，也可以是外部系统。

（3）一个用例可以有不同粒度，但一个用例要是一个完整使用场景，不要把单个步骤作为用例。

（4）用例之间有泛化、包含和扩展关系。泛化适用于表示一个抽象用例的多种不同实现。包含是一个复杂用例的多个细粒度用例。扩展表示一个用例的扩展能力，没有这个扩展用例原用例还是完整的。

（5）对用例图中的各个用例进行详细说明，从使用者的角度说明如何使用系统，不用描述软件系统内部如何实现该功能

## 业务流程

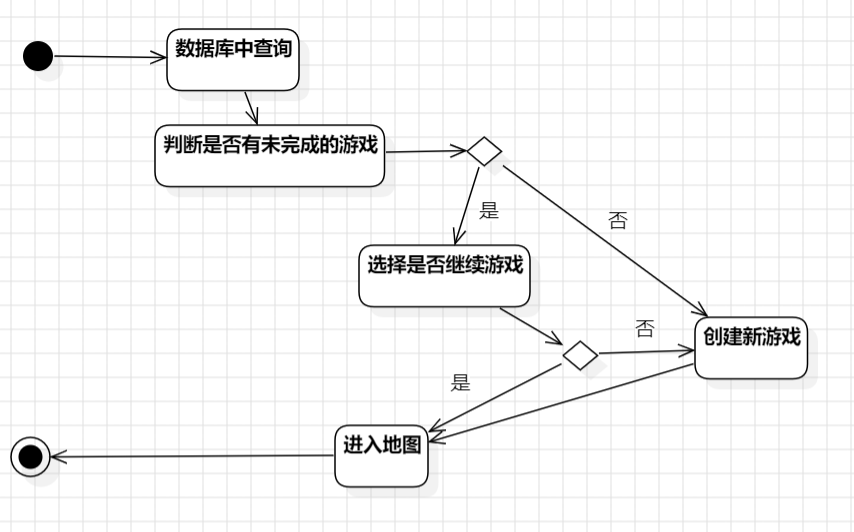


图 2开始流程图

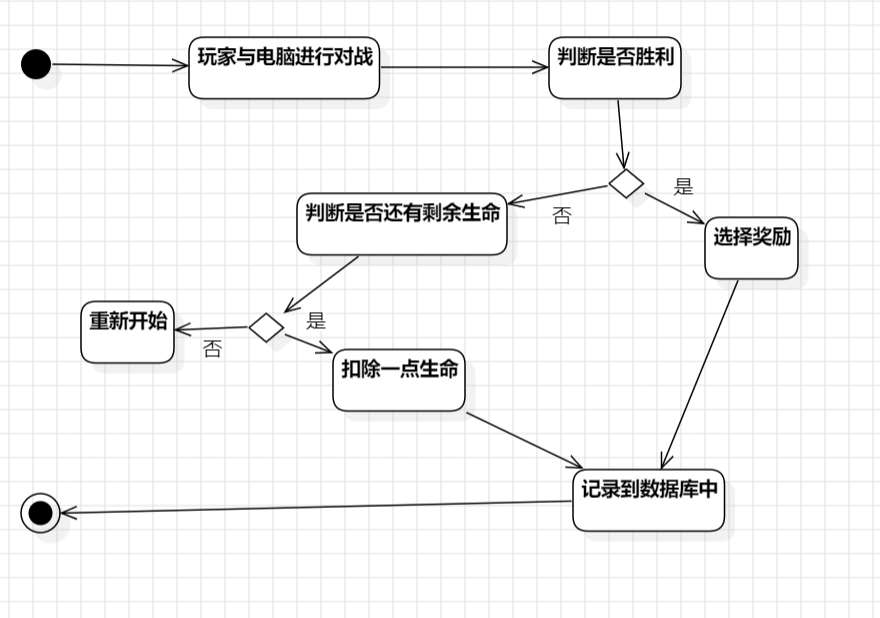


图 3对战流程图

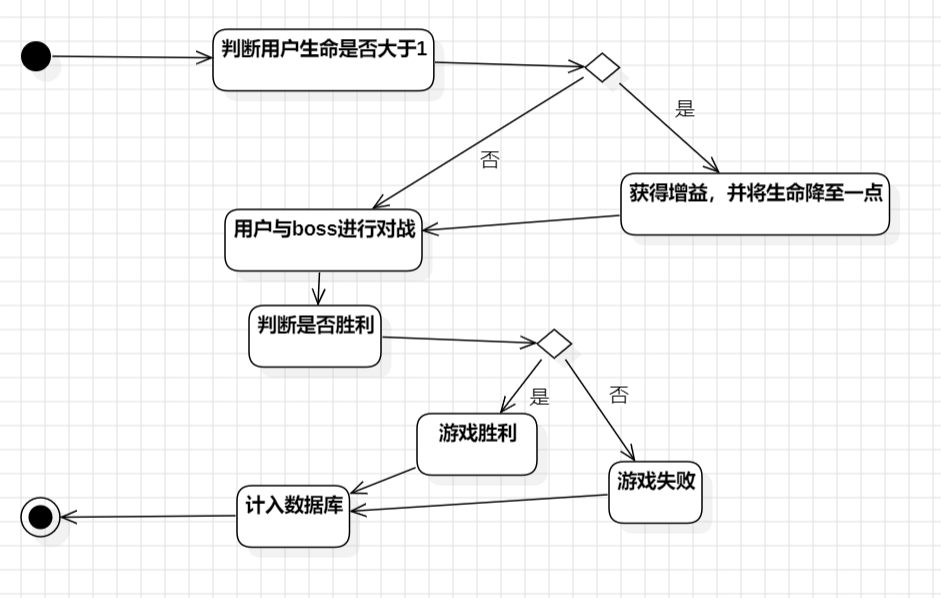


图 4 boss战流程图

说明：

1. 使用UML活动图（使用StarUML或华为的CodeArts Modeling绘制）画出系统的使用和操作过程。
2. 不用画出所有的使用场景，画出一两个关键用例的使用过程即可
3. 在需求阶段，活动图可以画系统做了什么，但不用画系统内部如何做的。

## 概要设计

## 功能模块设计

菜单 存档加载、设置

卡牌系统 卡牌生成、属性配置、组合效果计算

战斗系统 回合控制、伤害计算、状态异常处理

叙事系统 分支选择存储

数据管理 卡牌收集进度，卡牌数值、标记记录

​

说明：

1. 对各个功能模块需要完成的事情进行简要介绍。
2. 注意功能模块图和用例图不同，这里偏重于软件实现需要做功能模块，有些模块可能用户看不见，但也要画处理。
3. 按照功能来划分模块，不要按照前后端来划分模块。一个模块可能包括前端也包括后端功能。功能模块和代码结构并对应。
4. **叙事系统**

**输入：**路线分支

**输出：**下一个路线节点事件

**功能概述：** 玩家点击不同路线上的节点进入不同的事件（战斗、抽卡、铭刻、融合等），并记录玩家的路线选择

1. **抽卡**

**输入：**点击某张卡牌

**输出：**加入玩家卡牌包

**功能概述：** 抽卡事件为路线节点之一，玩家抽三张卡选择一张加入手牌

1. **铭刻**

**输入：**两张卡

**输出：**一张卡获得令一张卡的印记

**功能概述：** 铭刻事件为路线节点之一，玩家牺牲掉一张卡，将此卡的特殊标记附加在令一张卡上。

**//其余路线节点不再详细解释**

1. **战斗系统**

**输入：**手牌

**输出：**造成、受到伤害

**功能概述：** 负责战斗逻辑

开始 → 抽牌阶段（第一回合跳过） → 资源生成（道具） → 出牌阶段 → 效果结算 → （胜利判断）

→ 敌方行动 → 状态刷新 → （失败判断） → 结束/重复

## 核心类图

说明：

1. 类图可以使用StarUML或CodeArts Modeling绘制
2. 画出系统中的关键类以及相互之间的关系。这里关键类包括实体类、控制类和边界类三种。实体类描述系统中使用的数据，控制类描述系统的操作和业务过程，边界类描述系统与外界（如用户或数据库）的交互。

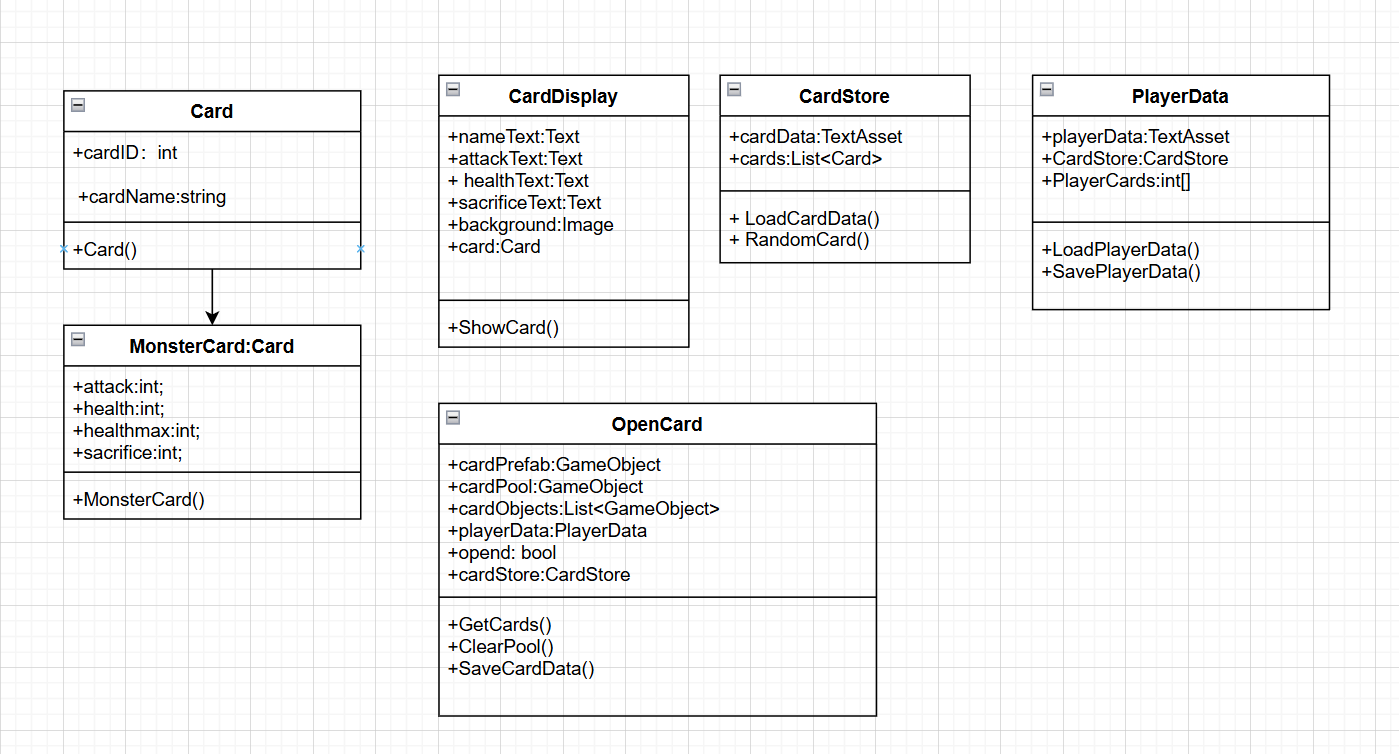


图 5核心类图

## 界面设计

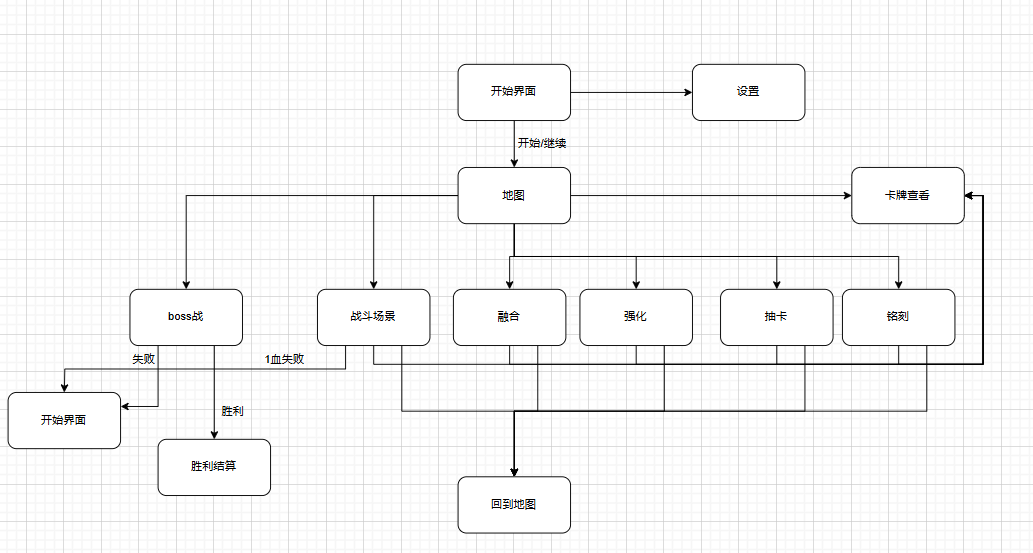


图 6界面设计图

说明：

1. 页面流程图：画一个系统的页面及其如何相互跳转的
2. 界面原型：画出系统主要操作界面，手绘示意图即可, 也可以用墨刀在线画更正式的UI原型