|  |
| --- |
| StudioL |
| Marisa Strike 系统设计文档 |
|  |

|  |
| --- |
| LarryMario  2015/1/3 |

目录

[1 引言 1](#_Toc408141985)

[1.1 编写目的 1](#_Toc408141986)

[1.2 项目信息 1](#_Toc408141987)

[1.3 项目背景 1](#_Toc408141988)

[1.4 参考资料 1](#_Toc408141989)

[1.5 术语表 1](#_Toc408141990)

[2 总体架构设计 2](#_Toc408141991)

[2.1 需求简述 2](#_Toc408141992)

[2.2 处理流程 2](#_Toc408141993)

[2.3 层次结构图 3](#_Toc408141994)

[2.3.1 展示层 4](#_Toc408141995)

[2.3.2 游戏逻辑层 4](#_Toc408141996)

[2.3.3 数据访问层 4](#_Toc408141997)

[2.3.4 数据层 4](#_Toc408141998)

[3 模块设计 4](#_Toc408141999)

[3.1 总体设计 4](#_Toc408142000)

[3.2 游戏逻辑模块 6](#_Toc408142001)

[3.2.1 ControlCharacter 6](#_Toc408142002)

[3.2.2 GetItem 7](#_Toc408142003)

[3.2.3 ControlEnemy 7](#_Toc408142004)

[3.2.4 RecordData 7](#_Toc408142005)

[3.2.5 ModifyState 11](#_Toc408142006)

[3.2.6 AutoSave 12](#_Toc408142007)

[3.3 装备模块 13](#_Toc408142008)

[3.4 商店模块 14](#_Toc408142009)

[4 数据库设计 14](#_Toc408142010)

[5 项目目录结构 15](#_Toc408142011)

# 引言

## 编写目的

本文档是软件工程课程项目——2D横版过关游戏《Marisa Strike》的系统设计文档，内容包括对游戏系统设计的说明。

本报告撰写完毕后，由软件项目总监进行审查，由软件工程师进行阅读。

## 项目信息

* 项目名称：2D横版过关游戏《Marisa Strike》
* 项目提出者：LarryMario
* 项目开发者：LarryMario、ScottFoH、lizhen2013、柯林中将

## 项目背景

近年来，独立游戏越来越受到重视，优秀的独立游戏层出不穷。同时，相关的开发资源，即开发工具、开发教程、图像素材、音乐素材等也越来越多。同人游戏也是独立游戏的一个形式，本项目组即是想在这样一个环境下，尝试进行游戏开发，积累开发经验。

本游戏以《魂斗罗》、《合金弹头》等横版过关游戏为游戏模式原型，遵循《东方Project》的剧情相关设定，加入装备与道具系统，旨在制作出一款具有较强可玩性、重玩性的东方Project同人游戏。

本开发小组旨在通过这次游戏开发，积累一定的游戏开发经验，并为《东方Project》二次同人事业的发展做出自己的贡献。

## 参考资料

* Shari Lawrence Pfleeger, Joanne M.Atlee. *Software Engineering*: *Theory and Practice, Fourth Edition*
* GBT 11457-2006 信息技术 软件工程术语

## 术语表

* 场景：特指Unity引擎中的“Scene”。Unity以1个场景为基本运作范围。
* 游戏对象：特指Unity引擎中的“GameObject”，指使用Unity开发的游戏中的一个独立的对象，例如摄像机、一个方块、角色等。
* 部件：特指Unity引擎中的“Component”，指挂载在游戏对象上的功能性单位，可以赋予游戏对象特定的功能。

# 总体架构设计

## 需求简述

本项目为东方Project同人游戏——《Marisa Strike》，游戏类型为2D横板过关动作游戏。玩家能够控制游戏中主角移动、攻击。游戏的简单人工智能系统能自动控制敌人移动、攻击，游戏相关数据会得到记录，必要的数据会显示在游戏画面上。游戏提供装备系统，能够使主角做出不同的动作、使用不同的攻击。游戏提供商店系统，玩家可购买道具以在游戏中使用。游戏提供自动存档系统，在特定的时间点游戏会自动进行存档，记录当前的游戏进度及道具拥有情况。

## 处理流程

如图2-1。

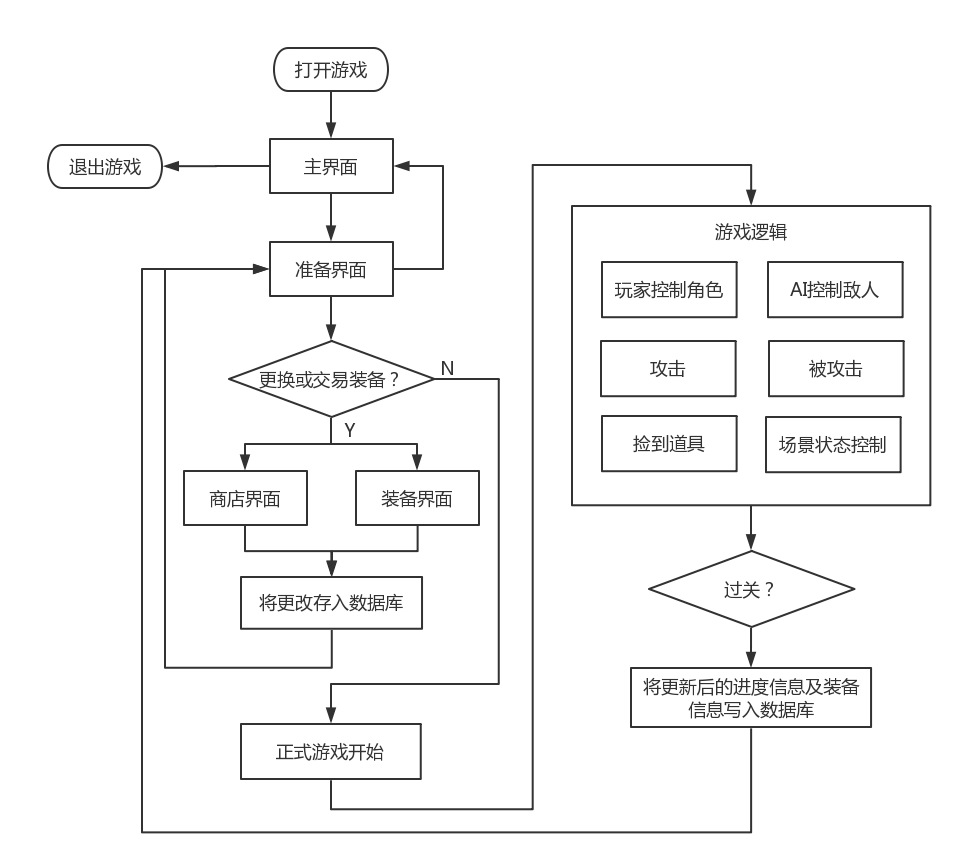


图 2-1 流程图

## 层次结构图

如图2-2。

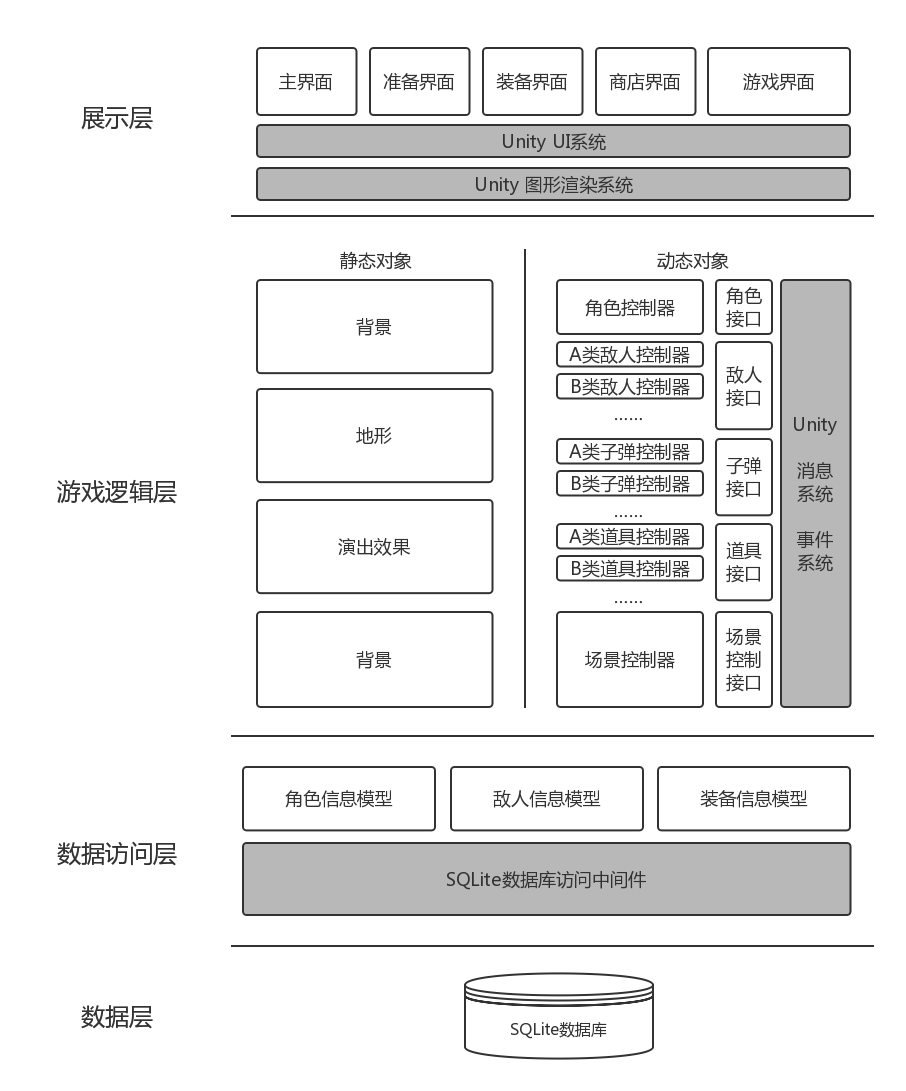


图 2-2 层次结构图

### 展示层

该层负责展示游戏画面，由若干场景构成，场景内已使用Unity编辑器放置好游戏对象，将通过Unity引擎的UI系统及渲染系统展示在游戏窗口中。

### 游戏逻辑层

游戏逻辑层负责对游戏进行控制，由若干控制器实体类、若干接口类与若干泛化类组成。控制器实体类分为两种：静态对象与动态对象。静态对象不会与其他游戏对象进行主动的交流，而动态对象需要与其他游戏对象进行通信，故必须作为接口类的实现。泛化类有两个作用：第一是泛化同类型的游戏对象，以便统一管理；第二是提升扩展性，以应对游戏所需要的多样性。接口类的作用在于利用Unity的消息系统与事件系统，使不同类型的实体类之间可以进行通信，以达到游戏对象间互动的需求

### 数据访问层

数据访问层负责在游戏逻辑层与数据层的数据传递中承上启下，由若干模型实体类，以及数据访问中间件构成。模型实体类负责承载来自上下两层的离散数据，组成有意义的实体，以供游戏逻辑层使用，或存储入数据层中。中间件则负责程序与数据库进行交流。

### 数据层

数据层负责存储游戏数据，由SQLite数据库构成。数据库均为离线形式，以文件方式存储在游戏玩家的游戏数据目录内。

# 模块设计

本章将以类图、时序图为展示方式，阐述模块设计。

## 总体设计

类图如图3-1与图3-2。

其中，IEventSystemHandler为Unity引擎所提供的接口，用途为与Unity的消息系统和事件系统进行对接。

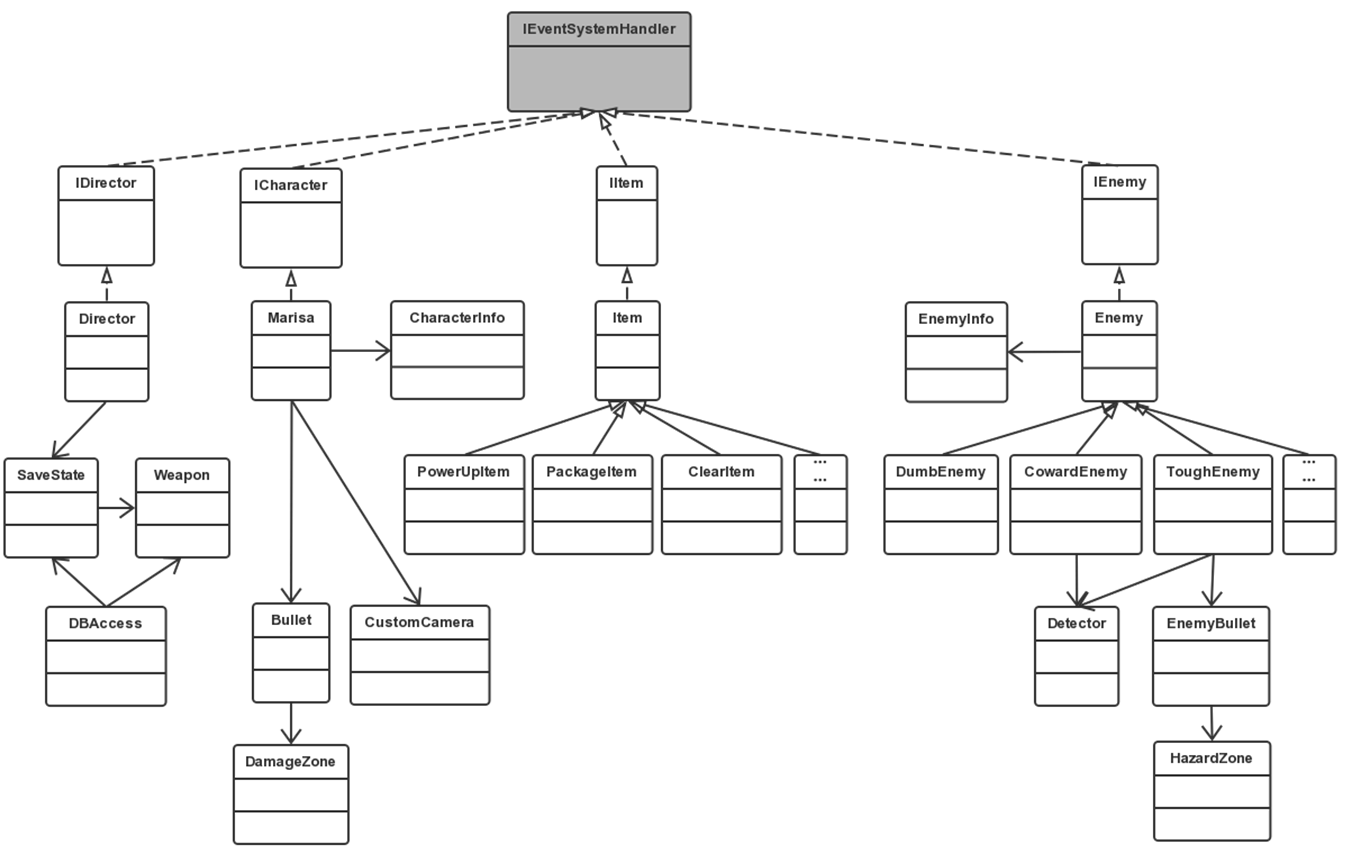


图 3-1 总体设计类图

## 游戏逻辑模块

本节将以用例为单位阐述设计。

### ControlCharacter

类图如图3-2，时序图如图3-3。

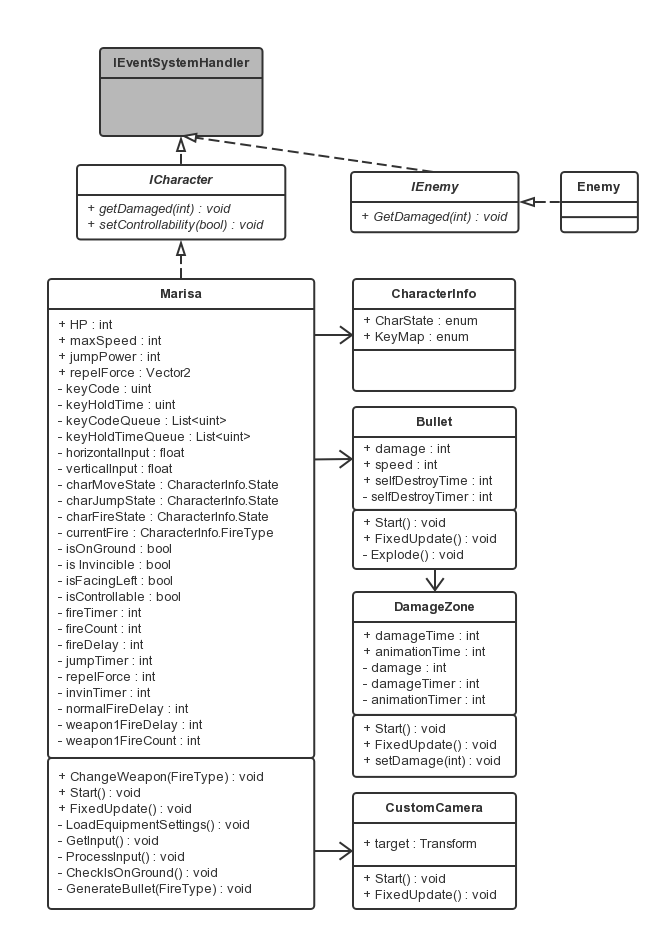


图 3-2 ControlCharacter类图

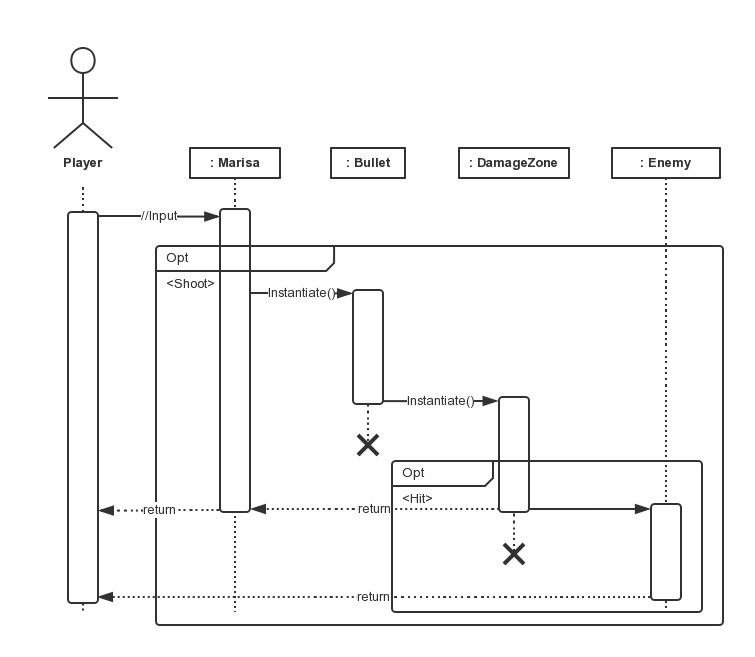


图 3-3 ControlCharacter时序图

### GetItem

类图如3-4，时序图如3-5。

类图中省略名字的类，代表今后可扩展的道具类型。

### ControlEnemy

类图如3-6，时序图如3-7。

### RecordData

类图如3-8，时序图如3-9。

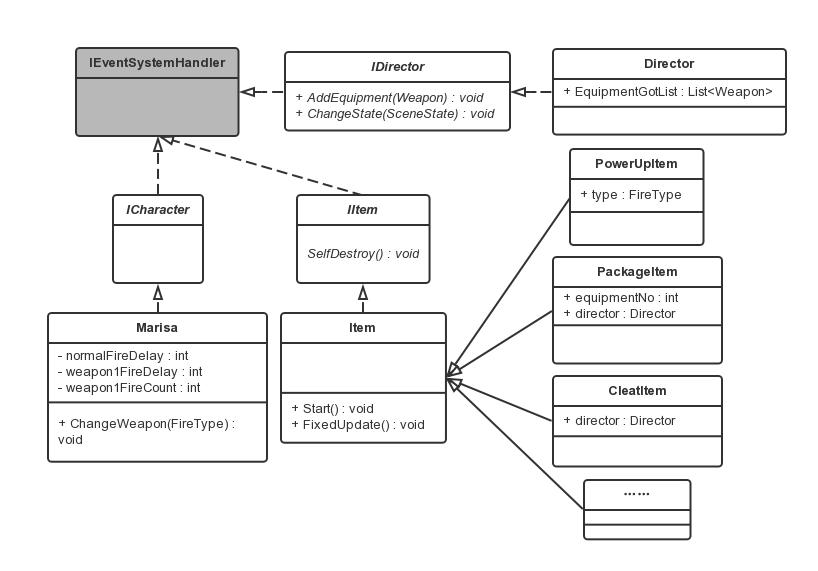


图 3-4 GetItem类图

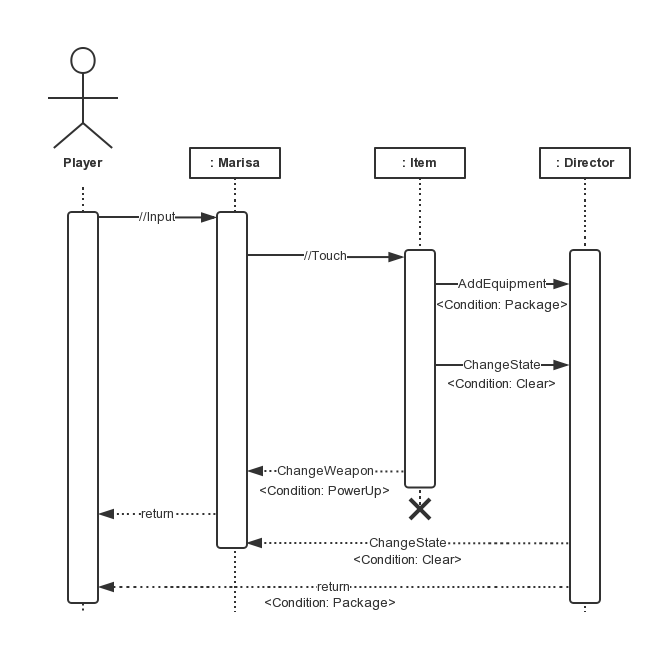


图 3-5 GetItem类图

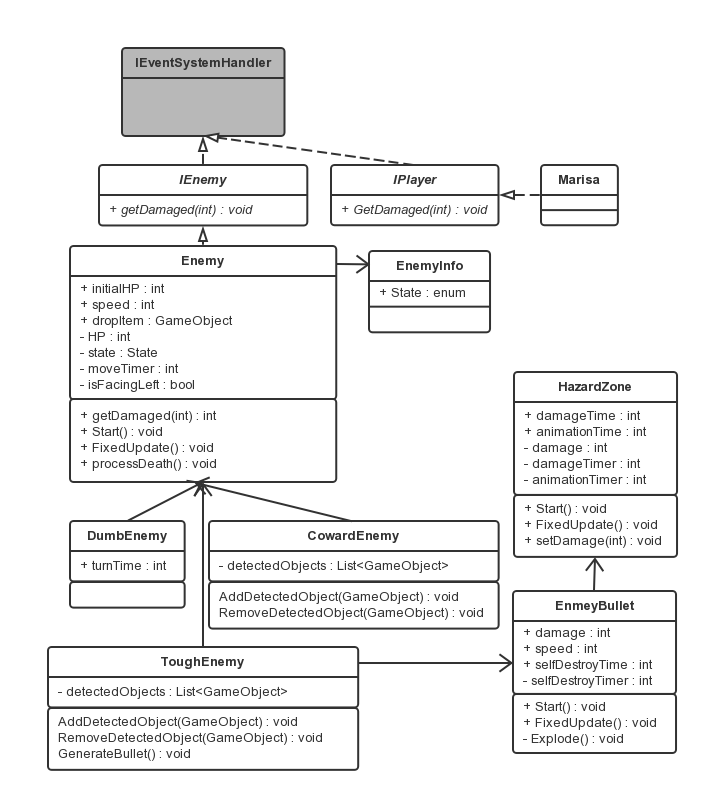


图 3-6 ControlEnemy类图

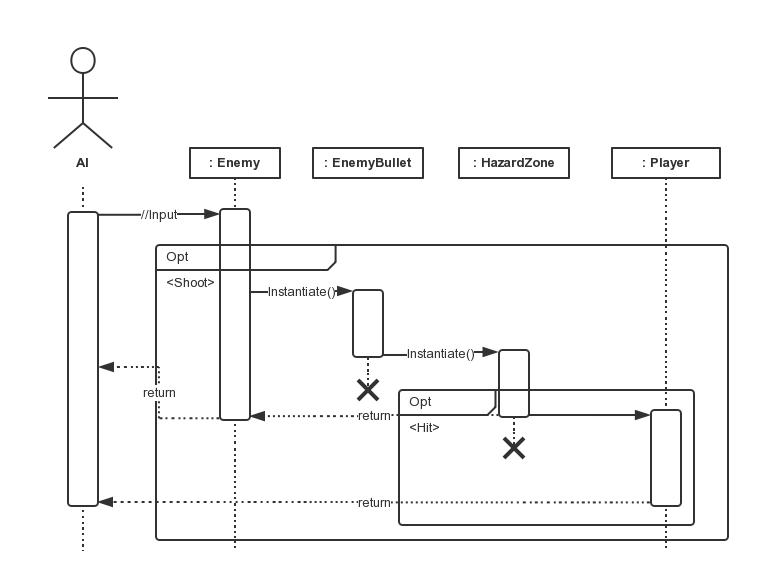


图 3-7 ControlEnemy时序图

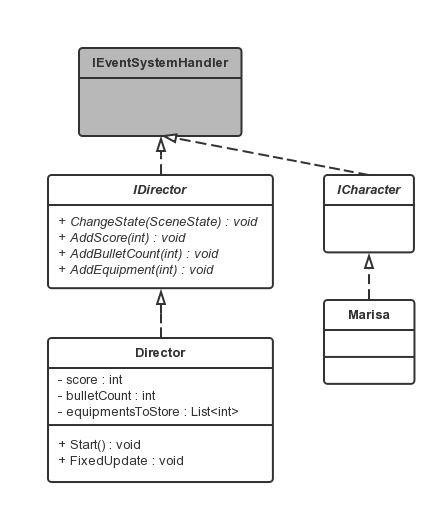


图 3-8 RecordData类图

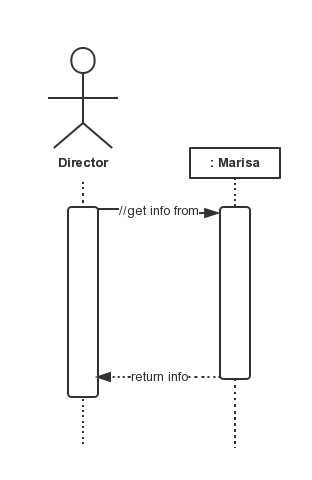


图 3-9 RecordData时序图

### ModifyState

类图如3-10，时序图如3-11。

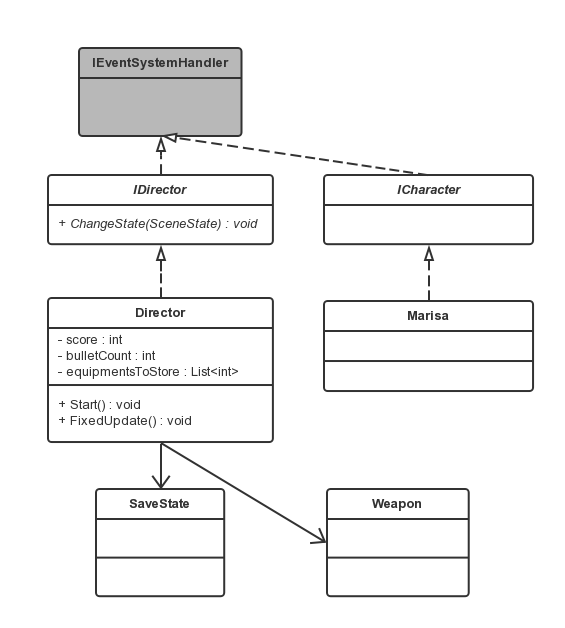


图 3-10 ModifyState类图

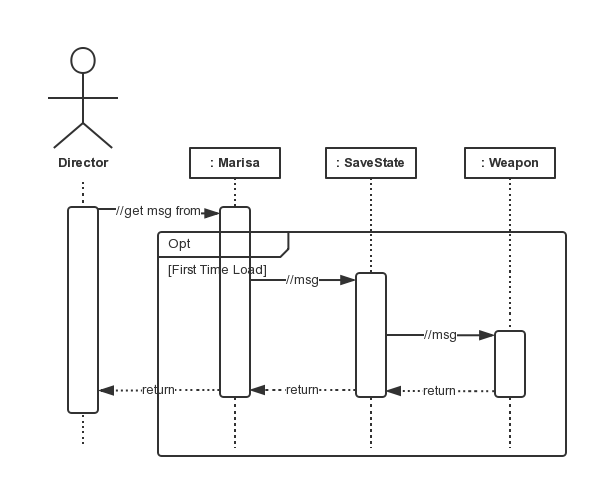


图 3-11 ModifyState时序图

### AutoSave

类图如3-12，时序图如3-13。

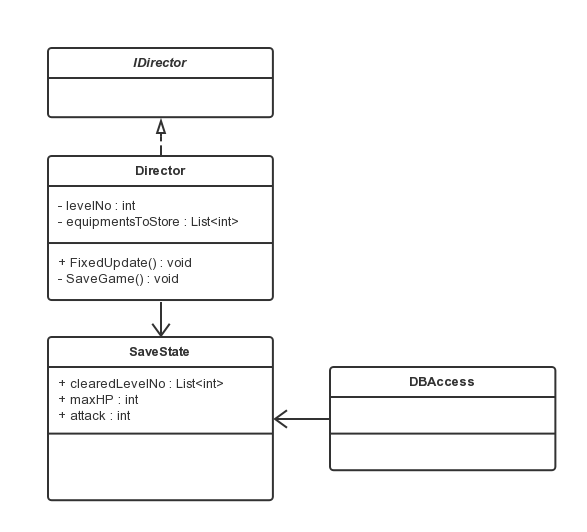


图 3-12 AutoSave类图

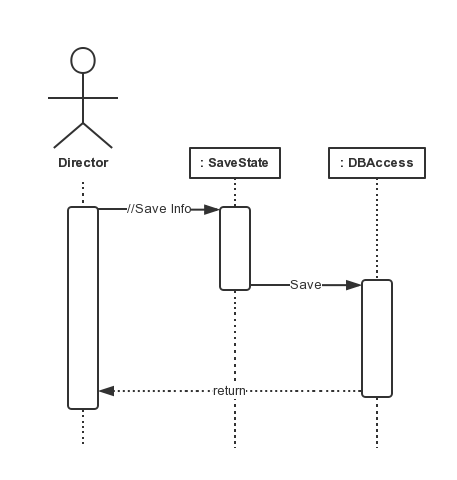


图 3-13 AutoSave类图

## 装备模块

装备模块的各个用例所涉及到的类基本相同，过程亦类似，故使用一张类图和一张时序图来阐述设计，如图3-14和图3-15。

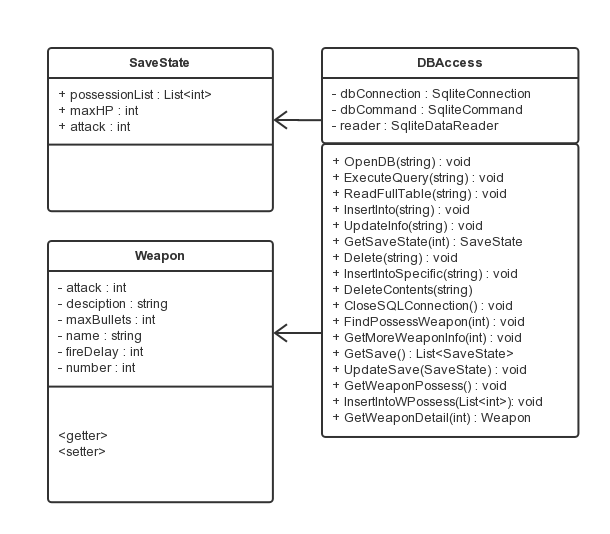


图 3-14装备模块类图

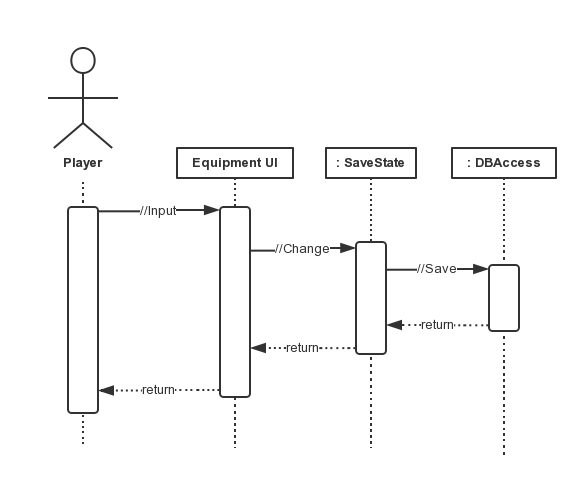


图 3-15装备模块时序图

## 商店模块

商店模块情况与装备模块类似，且涉及的类与装备模块相同，故类图可沿用图3-14。时序图见图3-16。

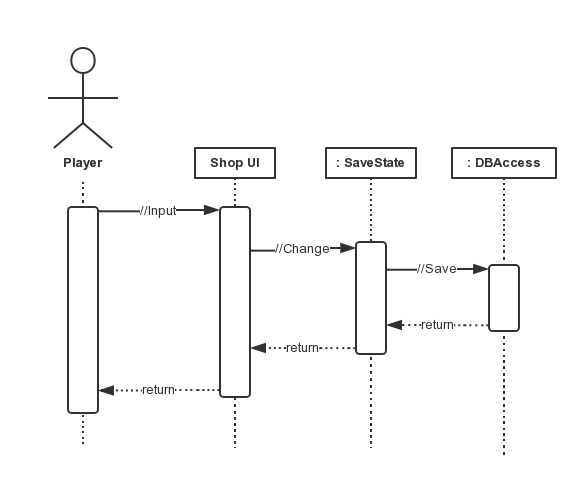


图 3-16商店模块时序图

# 数据库设计

由于数据库部分尚未实装到系统之中，故本章阐述的数据库设计尚为初步方案，需要进行进一步修改和完善。

数据库设计如下列各表。

## CharacterInfo

本表用作游戏存档。设计如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 非空 | 描述 |
| level | INT | 45 |  | 人物等级 |
| HP | INT | 45 |  | 人物血量 |
| weapon\_no | INT | 45 |  | 人物所持有的武器编号 |
| scene\_passed | INT | 45 |  | 已经闯过的关数 |

## WeaponList

本表记录所有武器信息。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 非空 | 描述 |
| weapon\_no | INT | 45 | 是 | 武器编号 |
| name | CHAR | 45 |  | 武器名 |
| attack | INT | 45 |  | 武器攻击力 |
| time\_interval | INT | 45 |  | 武器发射时间间隔 |
| max\_bullets | INT | 45 |  | 一次性最大发射子弹数 |
| description | CHAR | 100 |  | 武器描述 |

## WeaponPossess

本表记录已获得的武器信息。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 非空 | 描述 |
| weapon\_no | INT | 45 | 是 | 武器编号 |
| quantity | INT | 45 |  | 人物拥有的该武器数量 |

# 项目目录结构

参考本项目在Github上建立的在线仓库：

<http://github.com/larrymario/MarisaStrike>