拳皇开发文档

## 一、项目整体介绍

### 1、项目描述

拳皇是一款经典的格斗游戏，玩家需要控制自己的角色，击败对手。本项目基于Felgo框架开发，具有良好的跨平台性，支持iOS、Android、Windows等多个平台。玩家可以通过触屏或键盘操控角色。

游戏中有多个角色，每个角色都有不同的连招，玩家需要合理控制角色。

### 2、项目背景

暂无

## 二、研发任务分配

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **任务1** | | | | | |
| 开始时间： | | 2024.6.12 上午 10：30 | | 结束时间： | 2024.6.12 晚上 10：30 |
| **任务说明及分配** | | | | | |
| **编号** | **执行者** | | **任务详细说明** | | |
| 1 | 许 | | 分配任务，构建游戏程序初始类图，尝试熟悉felgo程序 | | |
| 2 | 梁 | | 制作开发文档，尝试熟悉felgo程序，收集素材 | | |
| 3 | 陈 | | 熟悉felgo程序逻辑，以及飞机大战游戏模型 | | |
| **任务成果** | | | | | |
| 1 | 创建了拳皇(KOF)仓库，初始化了开始界面 | | | | |
| **未完成任务** | | | | | |
| **角色移动呆滞，帧数不高** | | | | | |
| **编号** | **详细原因分析** | | | | |
| 2 | 素材质量不高，图片每帧大小并不等于实际角色大小 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **任务2** | | | | | |
| 开始时间： | | 2024.6.13 上午 8：30 | | 结束时间： | 2024.6.15 晚上 10：30 |
| **任务说明及分配** | | | | | |
| **编号** | **执行者** | | **任务详细说明** | | |
| 1 | 陈 | | 熟悉felgo语法，完善初始类图 | | |
| 2 | 梁 | | 熟悉felgo语法，整理资源，及时完善项目文档 | | |
| 3 | 许 | | 熟悉felgo语法，尝试重构初始版本拳皇 | | |
| **任务成果** | | | | | |
| **编号** |  | | | | |
| 2 | 基本完成，绘制出了第一个版本的初始类图 | | | | |
| **未完成任务** | | | | | |
| **重构初始版本拳皇遇到问题** | | | | | |
| **编号** | **详细原因分析** | | | | |
| 3 | 基础版本拳皇已不适合当下情形，已经重新分配任务 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **任务3** | | | | | |
| 开始时间： | | 2024.6.16 上午 10：30 | | 结束时间： | 2024.6.18 晚上 10：30 |
| **任务说明及分配** | | | | | |
| **编号** | **执行者** | | **任务详细说明** | | |
| 1 | 许 | | 分析并初始编写拳皇基本人物类 | | |
| 2 | 梁 | | 分析并初始编写菜单和游戏背景 | | |
| 3 | 陈 | | 分析并初始编写角色选择 | | |
| **任务成果** | | | | | |
| **编号** |  | | | | |
| 1 | 基本完成.实现了角色的选择功能 | | | | |
| **未完成任务**  **角色选择之后与之后游戏界面的角色初始化** | | | | | |
| **编号** | **详细原因分析** | | | | |
| 1 | 无法传递选择界面已经选择的角色实参到战斗界面之中正确地创建并初始化角色 | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **任务4** | | | | | |
| 开始时间： | | 2024.6.19 上午 10：30 | | 结束时间： | 2024.6.20 晚上 10：30 |
| **任务说明及分配** | | | | | |
| **编号** | **执行者** | | **任务详细说明** | | |
| 1 | 许 | | 分析并编写游戏主逻辑,完成游戏主功能 | | |
| 2 | 陈 | | 分析并编写启动界面动画和设置界面 | | |
| 3 | 梁 | | 分析并初始编写角色连招逻辑 | | |
| **任务成果** | | | | | |
| **编号** |  | | | | |
| 1 | 基本完成 | | | | |
| **未完成任务**  **游戏开场动画的实现** | | | | | |
| **编号** | **详细原因分析** | | | | |
| 1 | 与使用状态机进行界面切换的基础逻辑相冲突 | | | | |

## 三、问题分析处理

### 现阶段出现的问题

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 发现时间 | 问题描述 | 解决方案 |
| 1 | 6.12 17:30 | 合适游戏素材难找 | 暂无 |
| 2 | 6.14 16:30 | 控制人物移动无法翻转 | 暂无 |
| 3 | 6.15 | 滚动背景间出现空隙 | 将背景图边缘处叠加 |
| 4 | 6.23 | 如何进行角色之间通信连接 | 使用Qt的Network模块 |
| 5 | 6.22 | 如何使用九宫格角色选择界面 | 由于涉及到如何将选择的角色参数，传递到战斗界面的角色初始化实体类，最初打算是针对于本机的角色挑选进行一个选择与否边框颜色的不同进行区分，如1p选择之后边框变为红色，2p选择之后角色边框变为蓝色，由于涉及局域网连接，如何接受到对面选择的角色是关键，以及考虑到可能选择同一个角色的情况，决定修改为实时传递角色选择 |
| 6 | 7.3 | 不同Scene之间的合理通信 | 使用js封装一个共享库 |

### 2、可能出现的问题

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 发现时间 | 问题描述 | 解决方案 |
| 1 | 6.15 09:10 | 角色移动不流畅 | 使用物理引擎 |
| 2 | 6.24 | 角色碰撞检测不是很合理 | 对每种技能的碰撞进行单独检测 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 四、项目分析设计

### 1、游戏介绍

本项目是一款基于felgo开发的拳皇游戏，游戏玩法类似于经典的拳皇97，玩家需要操作自己的角色，躲避敌人的攻击并消灭敌人，最终打败对手。

游戏支持局域网联机模式，让玩家可以与其他玩家进行实时对战，体验更加刺激的游戏体验。

### 2、概要设计

概要设计如下：

1. 游戏简介：拳皇97是一款基于felgo开发的经典角色格斗游戏，玩家需要在众多的角色之中进行选择，操作自己选择的角色与对方的角色进行格斗对战，在各种操作连招之下，打出华丽的进攻，同时化解对方的猛烈攻势，在时间限制之内击败对方，赢得最后游戏的胜利。
2. 游戏模式：本游戏支持联机，在联机之中，玩家通过本地局域网进行连接 。进行PVP的华丽对战
3. 游戏界面：游戏界面采用精美的图形和音效，包括游戏场景、角色、角色技能连招、血条等，让玩家感受到真实的角色对战战斗体验。
4. 游戏操作：玩家通过手指滑动屏幕或按下键盘对应按键控制自己的角色进行移动和攻击躲避，击败对手。
5. 游戏地图：游戏设有多张地图，采用随机数设计，每次进入战斗时背景图都是随机产生，给玩家不一样的游戏体验。
6. 游戏设置：游戏设有设置选项，玩家可以自由设置音效、音乐、语言等。
7. 游戏更新：游戏将会不断更新，添加新的角色、地图和功能，为玩家带来更多的乐趣和挑战。

#### 2.1、游戏主要功能模块介绍

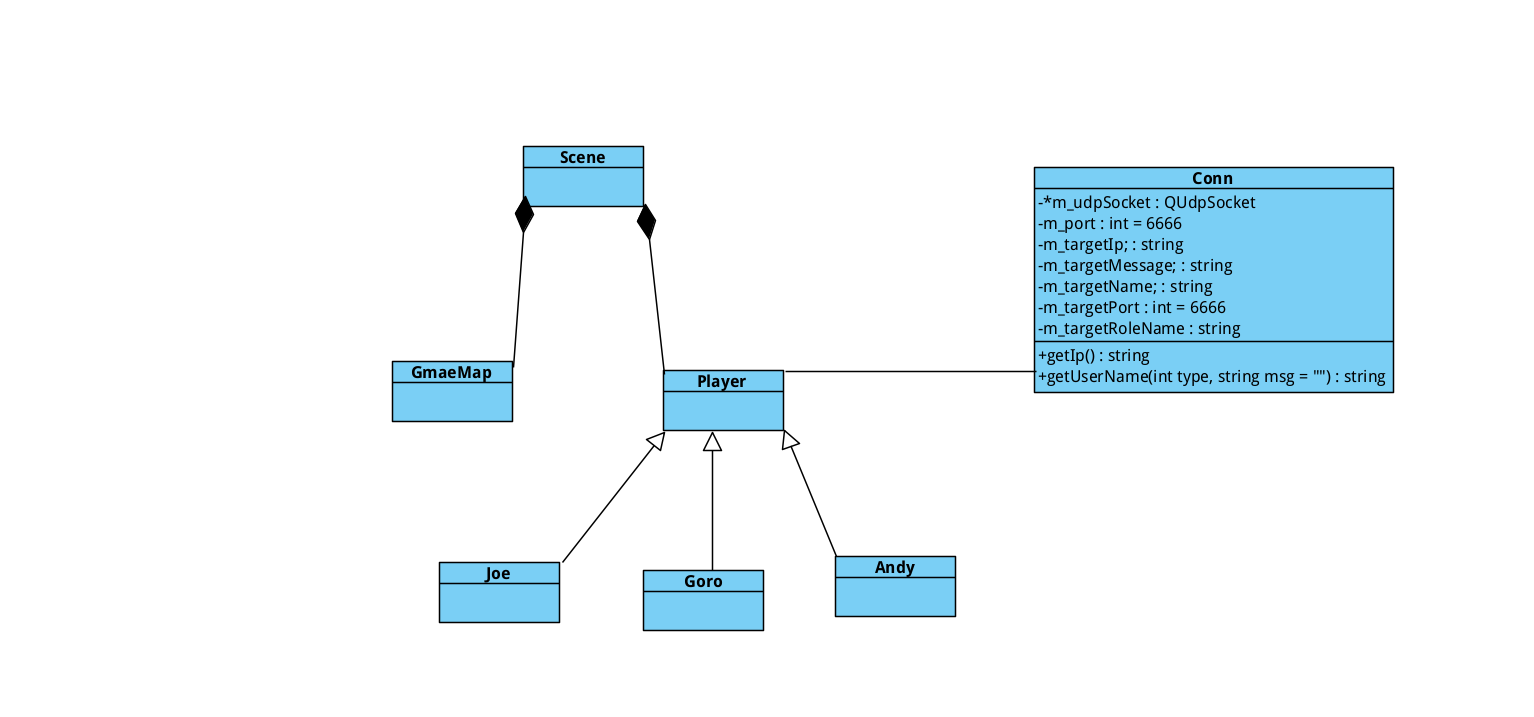
##### 2.1.1、游戏模块

开发拳皇游戏需要设计以下游戏类：

1. 角色类：包括所有游戏角色。
2. 地图类：包括所有地图。
3. 场景类：包括所有场景。
4. 网络连接类：用于辅助游戏玩家之间建立连接。

#### 2.2、 UML类图

##### 2.2.1、游戏逻辑初始类图：



### 3、详细设计

#### 3.1、介绍每个类的实现

##### 3.1.1、游戏模块

游戏模块分为三部分：界面（UI）模块、玩家（PLayer）模块和局域网连接（conn）模块

#### 3.2、设计思路及采用的先进思想

##### 3.2.1、检测方法：

角色之间的碰撞是使用矩形之间是否有交集，具体实现如下：

function is\_collision(r1, r2) {

if (Math.max(r1.x1, r2.x1) > Math.min(r1.x2, r2.x2))

return false;

if (Math.max(r1.y1, r2.y1) > Math.min(r1.y2, r2.y2))

return false;

return true;

}

##### 3.2.2、地图的多样化：

游戏的地图每次都会不同，具体如下：

import QtQuick

import QtQuick.Controls

AnimatedImage{

id:image

anchors.fill: parent

visible: true

source :"../../assets/img/bgc/" + getRandomInt(1, 23) + ".gif";

function getRandomInt(min, max) {

min = Math.ceil(min);

max = Math.floor(max);

return Math.floor(Math.random() \* (max - min + 1)) + min;

}

}

### 4、UI设计

#### 4.1、界面设计

##### 4.1.1、主程序画面：



##### 4.1.2、选择角色画面：



#### 4.2、运行平台及测试平台

felo 4.1.0 desktop for Qt 6.4.1

### 5、参考资料

—felgo官方游戏平台：[game.felgo.com](http://game.felgo.com)

—felgo程序中的官方文档

—CSDN平台中关于felgo程序开发的相关教程