

[宝可梦大乱斗]设计文档

1. 游戏介绍

本次 C++ 项目设计了一款游戏《宝可梦大乱斗》（Pokemon Fight），它是一款基于 Qt 的策略游戏。本游戏的主角来自于《精灵宝可梦》中的精灵，引入了等 17 个角色，它们在生命值、移速、攻击力、攻击范围等属性上各有不同。玩家需要在有限的能力点下选择最适合的宝可梦，并为自己的宝可梦设置阵型，以期让它们自己打败敌方队伍。

2. 游戏设计

2.1 界面设计

游戏开始后需要选择己方队伍：



图 1：选择队伍界面

点击 PLAY 进行阵型摆放，可以直接鼠标拖动宝可梦：



图 2：设置阵型界面

设置完毕之后，按“Enter”键进入游戏，宝可梦自动开始战斗。当敌方宝可梦全部倒下时，玩家胜利，可进入下一关卡：

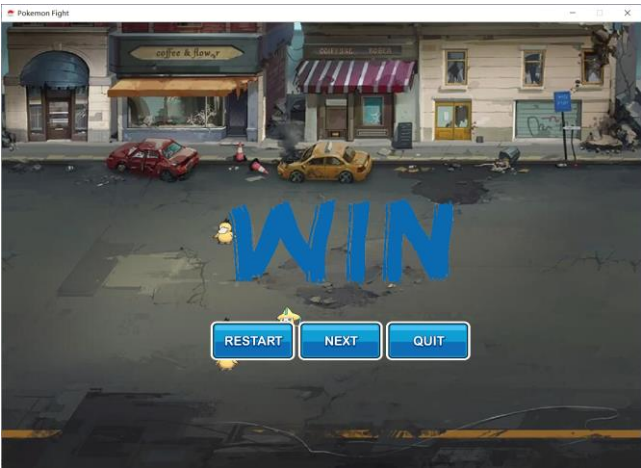


图 3：游戏获取界面

2.2 交互设计

在设置中可以关闭和开启声音。

在摆放阵型阶段，使用鼠标拖动宝可梦以移动位置。

战斗开始后，使用滚轮可以进行地图的放大和缩小。

战斗开始后，可以使用 Esc 键进行暂停。

（战斗开始后，可以使用 Ctrl+X 使己方宝可梦进入无敌状态。）

2.3 流程设计

内容：站在玩家的角度描述游戏的流程，模拟玩家的游戏过程画出流程图。

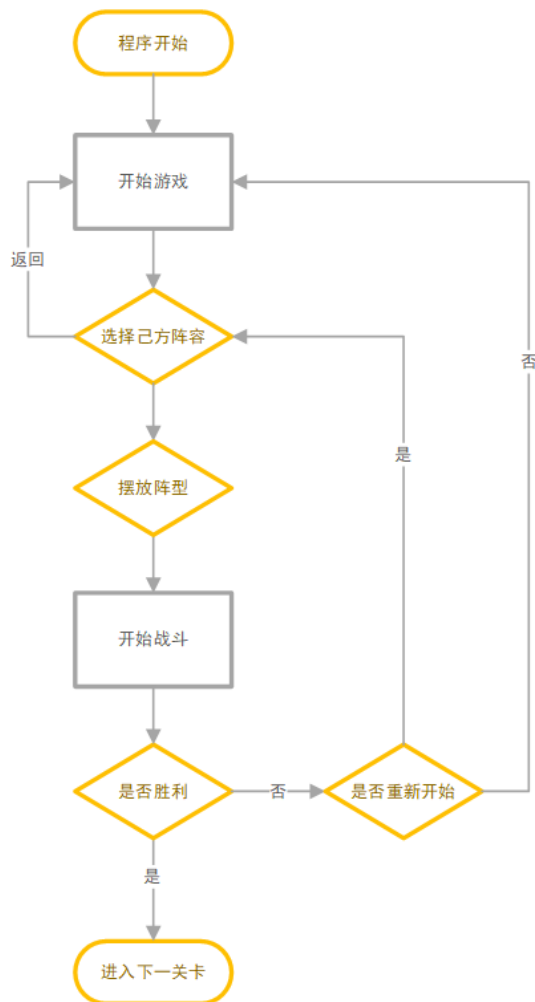


图 4：游戏流程

3. 游戏实现

3.1 代码流程

本游戏基于 C++ 和 Qt，使用面向对象思想进行设计。以下按照自底向上的顺序进行介绍。

首先定义一个类 `Pokemon`，类中定义了宝可梦的大小、移速、攻击力等属性，以及宝可梦的三种状态，实现了宝可梦的绘制、向指定位置移动等函数。另外，本游戏中的角色均以 GIF 的形式绘制，使得游戏中宝可梦的战斗和移动更加自然。

其次定义 `GameModel` 类，每个 `GameModel` 存储着每一局游戏的所有数据，也是游戏最核心的模块。`GameModel` 类中定义了敌方和玩家两个阵营（结构体 `Side`）的宝可梦，实现了增加各种类型宝可梦的函数，以及宝可梦寻找敌人、移动攻击的关键函数，控制游戏状态（暂停、继续）并且每隔一定时间检查游戏状态（是否胜利）。

最后是 `WarGame` 类，在这个类中控制着游戏的整体流程，如开始游戏（新建一个 `GameModel` 类）、设置、退出等。

此外，宝可梦的各种属性全部存储在 `pokemonability.h` 中，使用宏定义存储。

因此，本项目各个类和函数的作用分工明确，可扩展性极强。

总结如下：

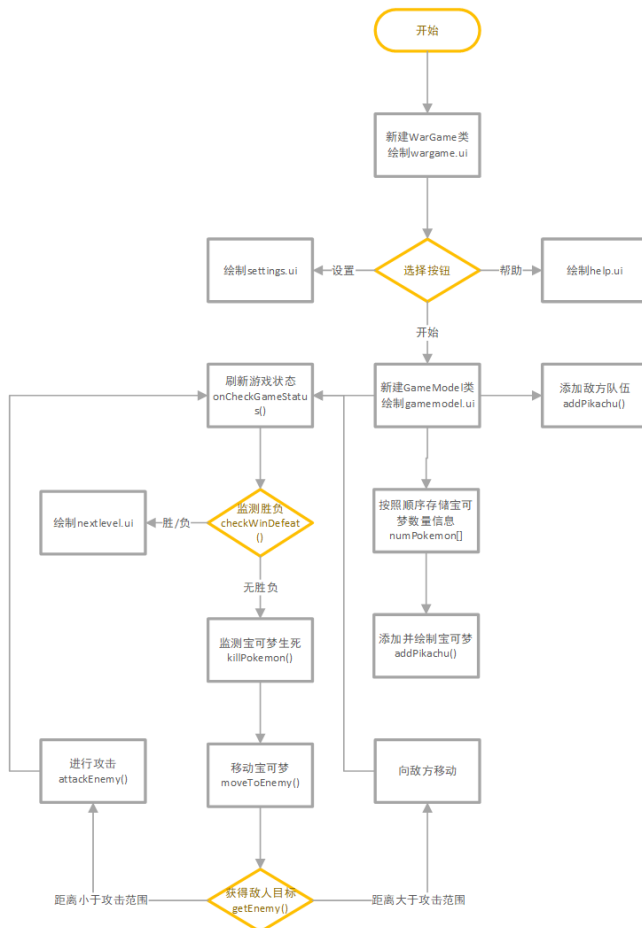


图 5：程序设计流程

3.2 核心模块

宝可梦添加和绘制机制：由于每个关卡的敌方宝可梦需要自定义，因此在 `WarGame` 中直接调用函数添加。而已方宝可梦需要玩家自己选择增删，因此使用了一个数组 `numPokemon` 按照顺序存储每种宝可梦的数量，玩家开始摆放队伍时，按照 `numPokemon` 调用相应的增加宝可梦的函数，并为其指定默认的位置。

移动和攻击状态机制：计算每个宝可梦与敌方宝可梦之间的距离，选择最近距离的宝可梦，如果距离大于攻击范围，则向敌方宝可梦移动，否则进入攻击状态。

移动机制：定时器每次刷新，宝可梦使用 `setPos()` 函数改变位置，改变方向为敌人位置，改变距离根据速度而定。由于定时器刷新很快，就产生了移动动画的效果。

攻击机制：每个宝可梦有自己的 HP 值，当宝可梦 A 因宝可梦 B 而进入攻击状态后，每隔一定时间，B 的 HP 值减少量为 A 的攻击力值。

4. 程序运行环境和步骤

从 Qt 官网 <https://www.qt.io/download> 下载 Qt Creator5.6 版本及其相关依赖，用 Qt Creator 打开项目，编译后即可运行。

5. 参考资料

[1] Qt 官方文档 <https://doc.qt.io/qt-5/index.html>