**第十一届中兴·图灵杯需求文档（修订稿二）**

**\*以下称东南西北时，均指绝对方向，即北为Z轴正向，东为X轴正向。**

**\*脚本中所有的变量、赋值、函数、循环、分支均需要做注释，每个脚本写完之后写一份接口说明。**

**\*文档中的变量说明内只对public变量进行说明。**

**\*本文档中，除非当前脚本在Update()函数内完成，否则对Update()函数均没有进行说明，请各位结合脚本功能对Update()进行编写。**

**\*本文档的函数说明仅为建议写法，各位可以自由发挥，但是请务必进行注释。**

**\*目前变量和函数均没有采取安全封装处理。**

**\*具体项目函数调用关系图请查看本文档同目录下的函数调用关系图.jpg**

1. **全局脚本**
   1. **GameManager脚本：**
      1. 脚本说明：

对游戏过程进行全局控制，包括游戏时间倒计时和游戏结束的分数显示。

* + 1. 变量说明
       1. 游戏剩余时间（public static int）
    2. 函数说明
       1. private void Timing()：对游戏时间进行倒计时。
       2. private void GameOver()：游戏结束，对其他脚本进行停用处理，使UI显示分数信息。
    3. 备注：

1. **地图脚本**
   1. **MapManager脚本：**
      1. 脚本说明：

控制地图的全局信息，地图信息以二维数组存储，-1表示不可用区域，0表示该处无方块，1表示该处有可炸方块，2表示该处有不可炸方块，3表示该处有炸弹。

* + 1. 变量说明：
    2. 函数说明
       1. public void UpgradeMap(int row, int col, int type)：通过穿进的值更新指定位置的方块类型。
       2. private void LoadFile()：加载地图 //TODO使地图成为Unity静态
       3. private void getMapInfo()：只读接口，返回当前全局地图信息（注意不要返回引用）。
       4. private int getBoxInfo(int row, int col)：智度解口，返回指定行与列的方块信息，值只能为-1，0，1，2，3。
       5. private void ReduceGameArea()：减小游戏区域（缩圈）
    3. 备注：

1. **方块脚本**

**方块占格1\*1\*1，有可炸和不可炸两种**

* 1. **BoxManager脚本：**
     1. 脚本说明：

每个地图方块均有一个实例，对当前方块的状态进行存储，以及本方块被炸时控制爆炸效果。

* + 1. 变量说明：
    2. 函数说明：
       1. private void setBoxType(int type)：设置方块类型。
       2. private int getBoxType()：获取方块类型。
       3. private void OnBoom()：执行爆炸效果。
    3. 备注：OnBoom()函数调用位置待考虑

1. **玩家角色**

a) 角色应有三个状态（Animator），原地静止，走动，死亡

b) 角色占水平面一格，高度两格

* 1. **PlayerMovement脚本：**
     1. 脚本说明：

用于控制角色移动，实现每次向角色前方移动定长1单位，向三个方向转向等。

角色移动或者停止移动时，在静止和走动状态间切换。

角色移动不能穿越任何方格，也不能穿越炸弹，但可以穿越其他玩家。放置炸弹时，角色可以和炸弹在同一个格子，但是不能从周围的格子走入已放置炸弹的格子。

每个角色都有跟随的摄像机。

* + 1. 变量说明：
    2. 函数说明：
       1. public void Movement（String toward）：使角色向指定方向（toward）移动一格。
       2. private void AnimChange(String stat)：控制角色动画状态
    3. 备注：
  1. **PlayerHealth脚本：**
     1. 脚本说明：

用于存储和控制角色血量并通知UI更新显示，初始血量为100，伤害为瞬间伤害。

血量小于0时，转换角色为死亡状态。

角色受到攻击时，身上发出红光并伴随提示音。

* + 1. 变量说明：
    2. 函数说明：
       1. public void getHP()：只读接口，获取当前角色血量
       2. public void TakeDamage(int Damage)：根据受到的伤害减少生命值（只允许为多个固定值防止此接口被利用）
       3. public void IncreaseHP()：增加一定血量。
       4. private void PlayerDeath()：判断玩家是否死亡，设置动画，禁用脚本。
    3. 备注：
  1. **PlayerShoot脚本：**
     1. 脚本说明：

角色向正前方发射子弹（可用Ray实现），攻击距离15格。不能击穿，只有击中玩家时才会造成伤害，瞬时伤害，-25血量，子弹飞行时间极短。

发射瞬间有火光和喷射火焰效果，可以用点光源和粒子系统实现。

发射瞬间播放开火的音频。

开火有间隔时间，相邻两发最短间隔时间为0.5秒。

* + 1. 变量说明：
       1. （仅用于测试）子弹基础伤害（public int）
       2. （仅用于测试）开火间隔（public float）
    2. 函数说明：
       1. public void Shoot()：对当前方向开火一次，若射中玩家，则扣除被射玩家一定血量。关联函数：PlayerHealth.TakeDamage(int Damage)
       2. public void IncreaseBulltDamage()：增加子弹伤害。
    3. 备注：多次测试血量和CD时间
  1. **PlayerBomb脚本：**
     1. 脚本说明：

角色可以在但前位置放置一个炸弹。

玩家每次只能放置一个炸弹，冷却时间2秒。

* + 1. 变量说明：
       1. （仅用于测试）放置炸弹冷却时间（public float）
    2. 函数说明：
       1. public void SetBomb()：在当前位置设置炸弹，创建炸弹GameObject。注意传递炸弹爆炸范围变量。
       2. public void IncreaseBombArea()：增加爆炸范围。
       3. public void reset
    3. 备注：
  1. **PlayerUpgrade脚本：**
     1. 脚本说明：

角色可以通过本脚本扣除一定积分，对自己进行短时间的技能升级或者回血。包括伤害复原。

* + 1. 变量说明：
       1. （仅用于测试）升级持续时间（public float）
       2. （仅用于测试）技能积分花费（public int）
    2. 函数说明：
       1. public void IncreaseHP()：提升血量。关联函数：PlayerHealth.IncreaseHP()、PlayerScoreManager.Upgrade()
       2. public void IncreaseBombArea()：提升炸弹爆炸范围，并在一定时间后复原。关联函数：PlayerBomb.IncreaseBombArea()、PlayerScoreManager.Upgrade()
       3. public void IncreaseBulletDamage()：提升子弹伤害，并在一定时间后复原。关联函数：PlayerShoot.IncreaseBulltDamage()、PlayerScoreManager.Upgrade()
    3. 备注：
  1. **PlayerScoreManager脚本：**
     1. 脚本说明：

对当前角色的分数进行存储和修改。

* + 1. 变量说明：
    2. 函数说明：
       1. public int getScore()：只读接口，获取当前积分值
       2. public void GainScore()：获得积分。
       3. public void Upgreade()：扣除一定积分，执行升级。
    3. 备注：

1. 炸弹

一个炸弹必须与放置它的玩家相联系

* 1. **BombManager脚本**
     1. 脚本说明：

炸弹将在定时1秒之后爆炸。爆炸瞬间，位于炸弹东西南北四个方向一个格子范围内的可炸方块将被炸掉，在上述范围内的玩家受到攻击-50血量。

炸弹爆炸时，有火光和粒子效果，并有爆炸音。

* + 1. 变量说明：
    2. 函数说明
       1. private void OnBooming()：炸弹爆炸，摧毁方块，销毁炸弹GameObject，增加玩家分数，扣除玩家血量。关联函数：BoxManage.OnBoom、PlayerHealth.TakeDamage()、PlayerScoreManager.GainScore()
    3. 备注：

1. 界面

界面显示四个玩家的血条、积分数。如果有附加技能，则显示附加技能图标，玩家死亡时将其头像置黑白。

如下图所示，进入游戏平台后显示**开始界面**，可以从指定文件中读取选手队名，选手所选择的人物（选手提供队名和人物编号，四个编号对应四张图片）。点击开始按钮进入游戏。

进入游戏后显示**比赛界面**或**单机演示界面**（单机演示界面可以先不做）。比赛过程中，若选手角色死亡，将对应队伍的角色头像置黑白。血槽需根据玩家血量实时更新。留有接口，当玩家获得附加属性时调用该接口，在相应位置显示出技能图标。

当场上剩余最后一名玩家时，游戏结束，显示**比分界面**，延时等待5秒之后反回开始界面，进行下一场游戏。

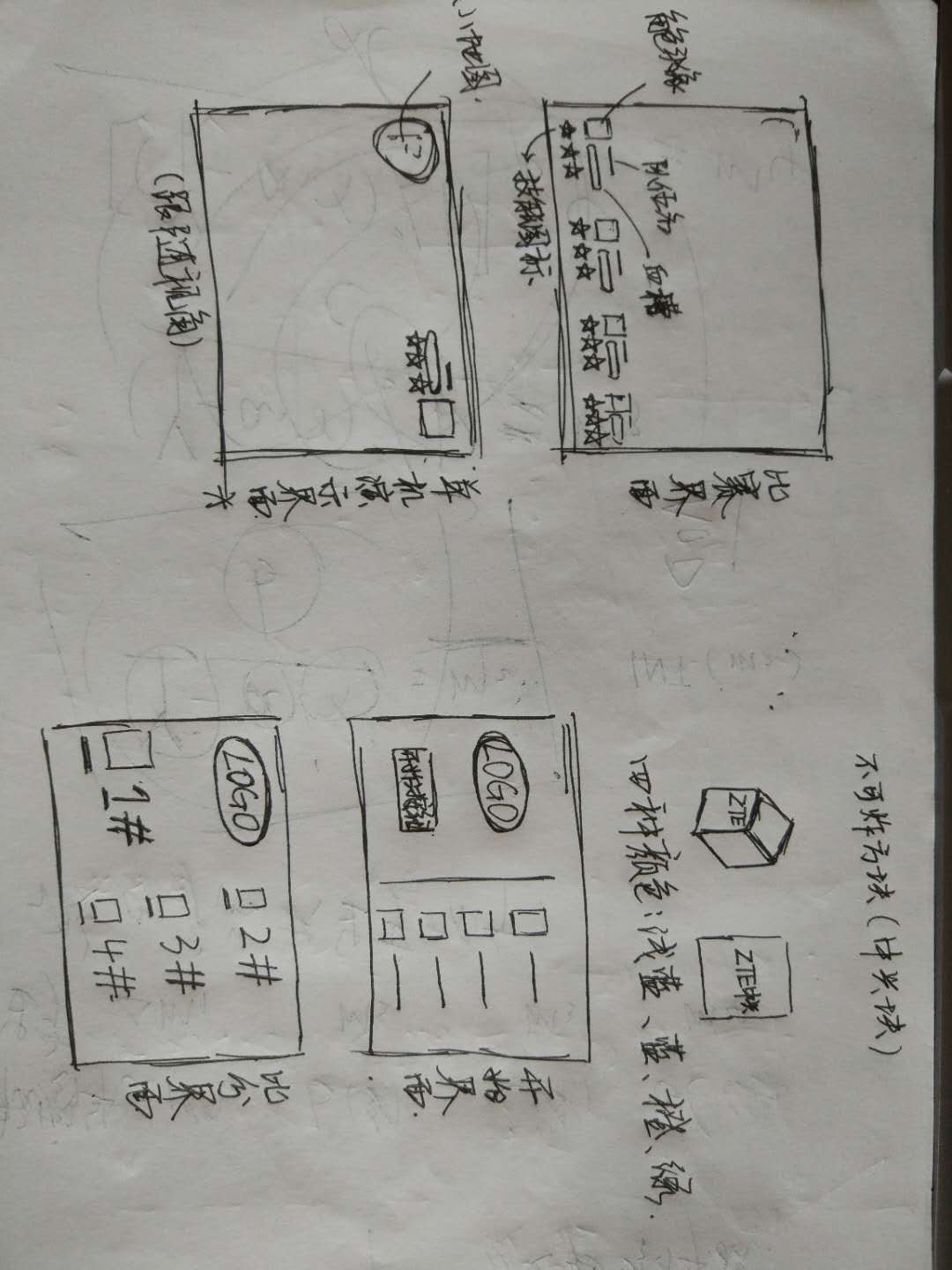


图6-1 比赛界面示意图

1. 玩家类封装
   1. 提供的操作有：朝四个方向的前进，开枪，放置炸弹，兑换某个附加属性
   2. 提供的信息有：全局地图信息数组，其他玩家的坐标（二维数组下标），其他玩家的血量，其他玩家获得的附加属性，游戏倒计时时间
   3. 存储玩家ID，队名
2. 其他操控方式接口预留

对于玩家类封装的几个操作接口，设置对应键盘操作接口

1. 测试