# Bunker Treasure Report

## 游戏概述：

游戏的名字是《Bunker Treasure》，是一款2D像素解谜游戏，Cyberpunk和神秘风格。欲发布平台是PC、Android和IOS，第一版需要开发大约10关。

### 游戏背景:

故事发生了20XX年的Bunker基地，入侵者植入了许多智能机器人到Bunker基地，要偷取核心能量石，玩家既是主角，一个半机械半人类的守卫者，通过完成任务升级自己的技能，也可以收集能力碎片，抵挡外来入侵机器人。

### *游戏玩法：*

是玩家控制一个最多10X10格子场景的横竖变换。一方面阻止外来入侵者偷取Bunker中的Treasure，另一方面通过拼接和消除场景中的特殊格子来获得力量或者完成任务。获得的力量分为道具类和被动能力类。且道具又分为无目标使用类型，和指定目标使用类型。

## 创新内容：

1. 对Cyberpunk和远古神秘风格的像素绘制，以及像素动画的绘制表现
2. 对代码架构的创新，在Unity中尝试使用MVC的架构格式。
3. 对关卡的设计，以及入侵者的AI设计

## 开发工具：

开发引擎：Unity3D

开发语言：C#

开发IDE：Visual Studio 2017

像素绘制软件：Asprite

## 开发细节：

### 美术任务细节

作为一个美术风格为CyberPunk和神秘感觉的像素画来说，美术工作分为如下几个步骤：

1. 首先寻找资源，确定感觉
2. 其次量化美术素材，进行绘制
3. 美术资源分层，输出序列帧
4. 整合到Unity3D中。

### 程序任务细节

作为程序框架结构的创新尝试，本项目的程序是这样架构的。

#### 新框架概述：

框架分为3个模块：BasicProcess，BasicModule和Unity本身的GameObject。其中GameObject作为“View” 模块，BasicProcess作为“Controller”模块，而BasicModule作为“Module”模块，他们三者的关系是：

管理

Module Manager

Module

更新GameObject

操作Module

事件和属性注册到Module

管理

GameObject

Process Manager

Process

其中，作为“View”模块的GameObject，会将其节点和event注册到Module中，在Module中，进行逻辑层运算，更新GameObject的相关组件和属性。然后Process中调用工厂模式ModuleManager操作Module生成，销毁等，同时为了与Unity框架相对应，Process与Scene相呼应，通过Process来切换Scene，管理场景中逻辑的相关事务。ProcessManager用于Process的相互切换。

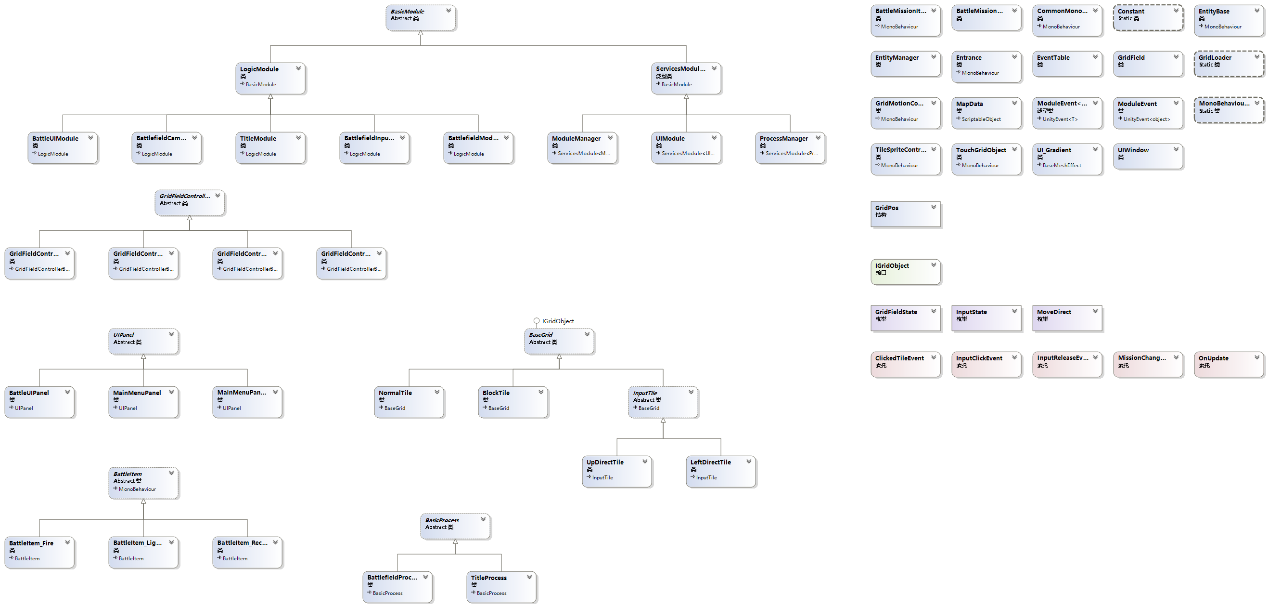
为了灵活使用Module模块，框架中又将Module派生出一个单件ServicesModule和一个LogicModule。Module通过C#反射机制，动态的Activator.CreateInstance出Module实例。

#### 新框架的优势：

传统的Unity3D开发中，程序员会根据Unity的GameObject和Component的设计方式。写很多继承自Monobehavior的各种功能的组件。在场景中，静态预先挂接到GameObject上，或者动态AddComponent到GameObject上。但是，这么做有一些弊端。对于协助完成游戏模块或者系统，在后期分析或者迭代项目时，无法理清多层级GameObject下挂接的众多组件之间的关系，也很难分析Component间相互的纠缠和耦合。所以，在Unity3D中尝试用MVC是一个很好的解决方案。

对于开发的程序员而言，清晰的结构更能显示出此框架的优势所在。在模块测试与调试阶段，由于逻辑间的低耦合，会使开发进度提升。而且在UI模块上，进行了高度封装，UIPanel和UIWindow，分别管理正常UI的GameObject和类似于模态对话框的UI。

类结构图如下：

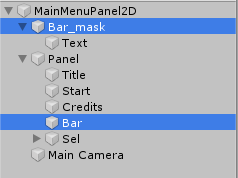


### 项目拆分

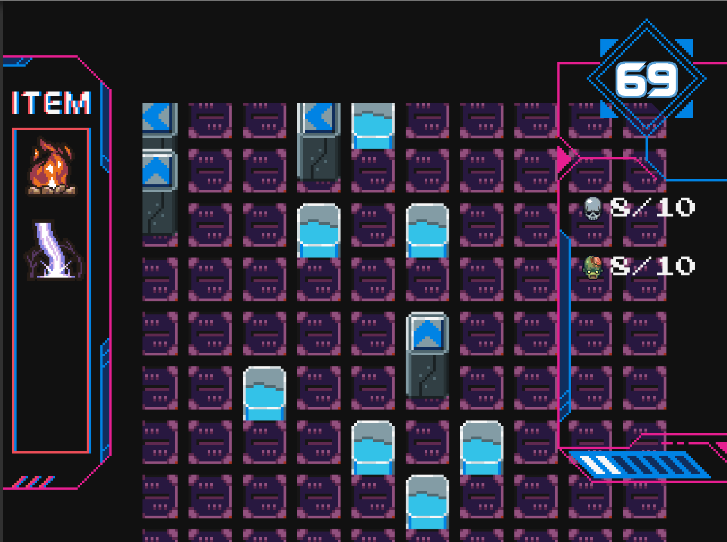
1. 项目为2D游戏项目，场景分为5个，Entance，LogoScene，MainMenu，MainGame，EndGame
2. Entance是入口场景，通过Entance.cs脚本来控制Manageres启动和初始化所有模块和Process，通过不同的Process来切换场景。
3. LogoScene是展示我们StudioLogo的过渡场景
4. MainMenu是主菜单页面，主要是UGUI交互和表现逻辑



通过UGUI的Mask和原图叠加制成条纹状滚动文字。



1. MainGame是游戏是主界面，效果图如下，分为如下几个部分：



1>左侧道具栏，支持拖拽滚动。

2>右侧关卡任务栏，右下为关卡进度条，右上为当前关卡序号。

3>中心为游戏操作界面，网格中特殊物品点击后支持横竖条带移动，且特殊物品相互靠近后可以双消或者三消。消除后给玩家提供力量。场景网格中会出现机器人，机器人会以一定的能力靠近场景中的能量宝石，玩家通过控制网格条带的移动，阻止机器人偷取能量宝石。

4>背景会有一些点缀用的特效或动画。

1. EndGame是游戏的结束界面，在当前关卡游戏完成之后（胜利或者失败），游戏会进入到EndGame的场景中，通过EndGame场景，玩家选择“Replay”“Next”还是“Quit”。

如果之后要进行商业化，可以在Replay和Next选项点击时，增加广告AD的播放，为开发人员提供一些微薄的收入。

游戏会在PlayerPrefs中记录游戏当前的关卡数和其他数据，在玩家再次登录游戏时可以读取。

## 结果和思考：