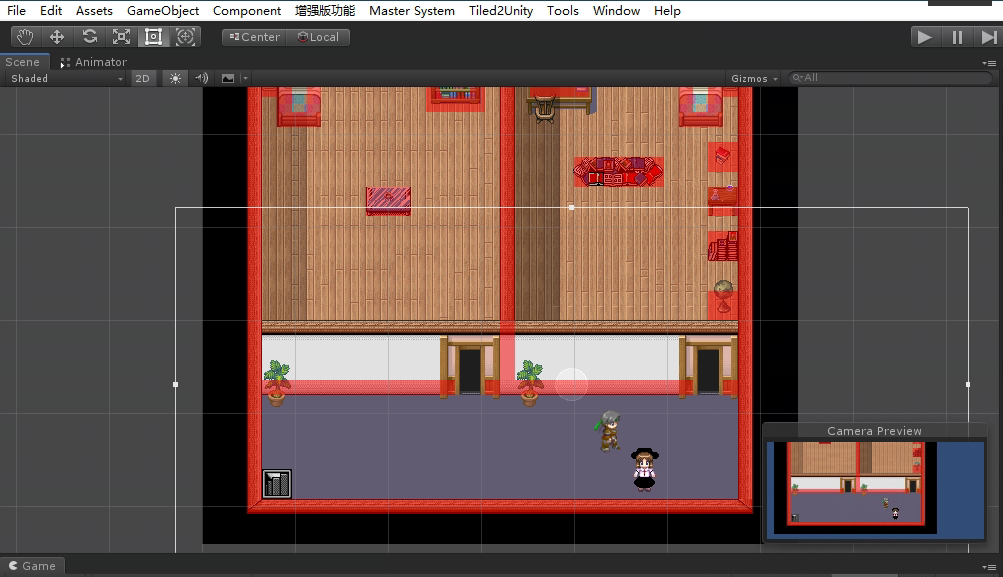
**一份个人对于游戏的阐述**

或许会有人好奇，为什么我要从物理专业转向游戏相关。其实在我看来，物理与游戏也是具有强相关性的。物理是对现实世界客观规律的发掘与总结，着重于发现。而游戏则是对虚拟世界的规律的定义，着重于创造，发现与总结则是玩家与游戏设计师之间的强强对抗。谁知道现实世界是不是一局广义的“游戏”呢？

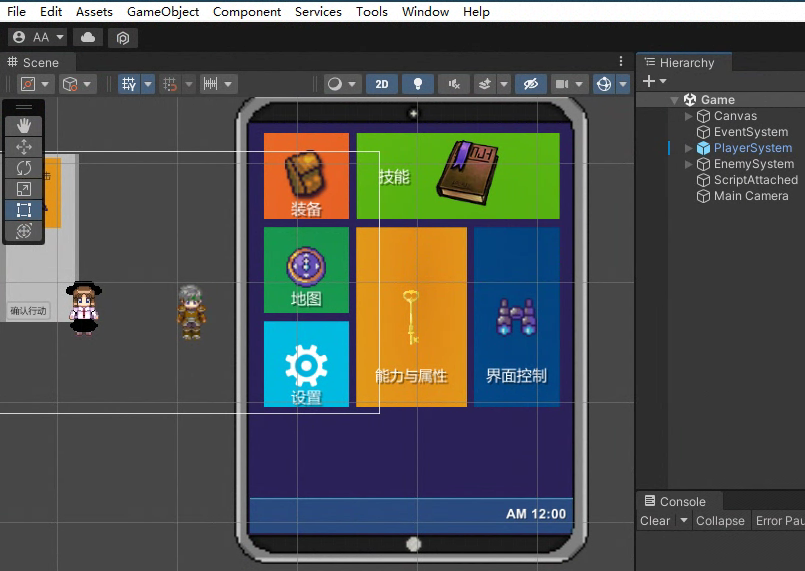
诚然，热爱游戏的最终目的归属于一款属于自己创做的游戏，至少对于我来说是这样的。这种动机来源于一款远古游戏《东方年代记》。这部作品隶属于东方project系列的同人创作，在继承原有的部分人物故事背景的设定下，制作团队为剧情加上了自己的巧思，最后结合RPG的游戏模式，为玩家奉献了一部具有引人入胜的经典同人作品。

我的动机非常简单，为什么不能自己也去制作一部类似的作品呢？（当然一开始的想法较为简单）在了解到《东方年代记》是以RPG maker 作为制作软件之后，我对RPG maker进行了一定程度的调研，RPG maker 虽然可以通过Java脚本作为支援，进行一定程度的自定义编辑，但其表现力和效率对标于Unity 与 Unreal 仍然占据不到优势。最后我决定使用Unity来作为实现我想法的工具。

在掌握的一般C#使用的前提下，我开始了unity的开发之旅。首先确定使用unity的2D模式，利用tiledMap 导入地图，先做出类似于原本游戏的场景：如图



对于UI,为了与主角（学生）身份相匹配，且为了增加游戏的代入感，我决定将所有的游戏操作合并到一个容器内:



在将游戏的一般场景初步搭建起来之后，便是最困难最复杂的步骤，游戏各类系统的搭建。在游戏运行的状态中，各模块应该动态加载，动态销毁。如怪物，技能特效等等，这些操作应该交给游戏中挂载的各类C#脚本自动完成。这样以来，在unity中从0开始搭建一个完善的游戏系统对于非计算机专业来说是十分困难的。在此我选择从B站或Github找到现成的游戏系统架构来直接使用。在此，UI系统通过运用B站某教程的搭建方式，成功实现了按键对鼠标的响应，经过部分修改后，UI系统具有了通用性，只要将脚本挂载在对象上，就可以实现通用的按键，滑轮滚动或拖拽操作。

下图是在运行时单击“装备”的情形：



并且通过单例模式，系统可以与外界沟通。对于技能系统，我运用了Github上的EplayGame框架，但此框架的通用性较强，代码内容繁多，目前还在尝试理解重构代码，将其修改为能实现回合RPG战斗的系统。

这个游戏目前看起来还是很残缺，但我想我会仍然会去给它注入我自己的想法，一点一点实现它。

对于游戏风格来说，我更偏向4X游戏或独立游戏，诸如P社系列，文明系列，或Undertale，Fast Than Light等小众向，在我看来4X游戏的系统丰富且有深度，非常适用于学习和变换创作，而独立游戏则频含亮点，非常适于启发想法。

以上就是一位非科班人的一些想法，在科班人看来或许很无趣。目前手头还存在一点科研任务，比较忙，没有时间将这篇阐述Latex化，格式没有太注重，也没有太注重内容，请见谅。游戏行业方兴未艾，在有更多新技术的加持下，是值得投入的。