# 2048小游戏

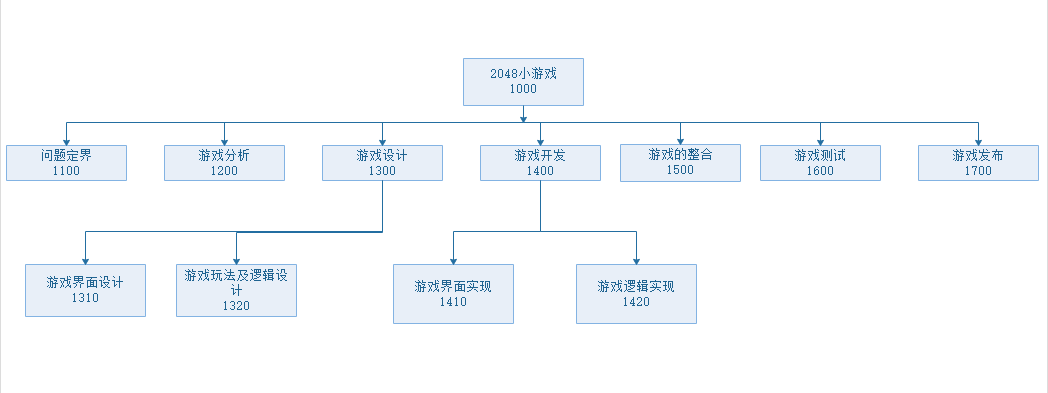
# 项目管理与计划书

姓名：孙阳

学号：SA18225329

### 项目分解：

1. WBS图：



1. 任务列表：



1. **规模估算：**

我们采用国际标准IFPUG组织提供的功能点估算法来对此2048小游戏进行规模估算。其公式为：

FP = CT\*[0.65+0.01\*∑Fi]

其中：

数据类型功能点：

内部逻辑文件ILF：用户分数信息文件：当前得分、当前合成最大数字、历史最高分、历史合成最高数字

外部接口文件EIF：本小游戏不含EIF文件

共1个ILF。

人机交互类型功能点：

用户修改当前得分、用户修改当前合成最大数字、用户修改历史最高分、用户修改历史合成最高数字、用户查询当前得分、用户查询当前合成最大数字、用户查询历史最高分、用户查询历史合成最高数字。

共4个EI，4个EQ。

计算EI复杂度：每个EI涉及到1个FTR，1个DET，复杂度低

计算EQ复杂度：每个EQ涉及到1个FTR，1个DET，复杂度低

计算ILF复杂度：涉及到4个DET，1个RET，复杂度低

因此根据未调整功能点复杂度对应表得CT = 1\*7+4\*3+4\*3 = 31

我们再根据图2.1求出Fi。

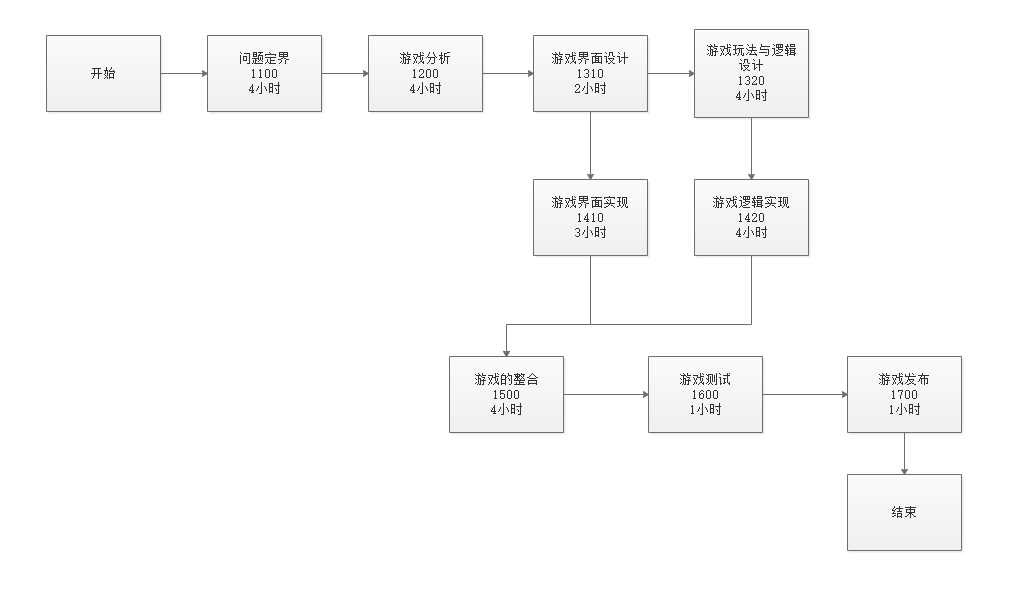


图2.1

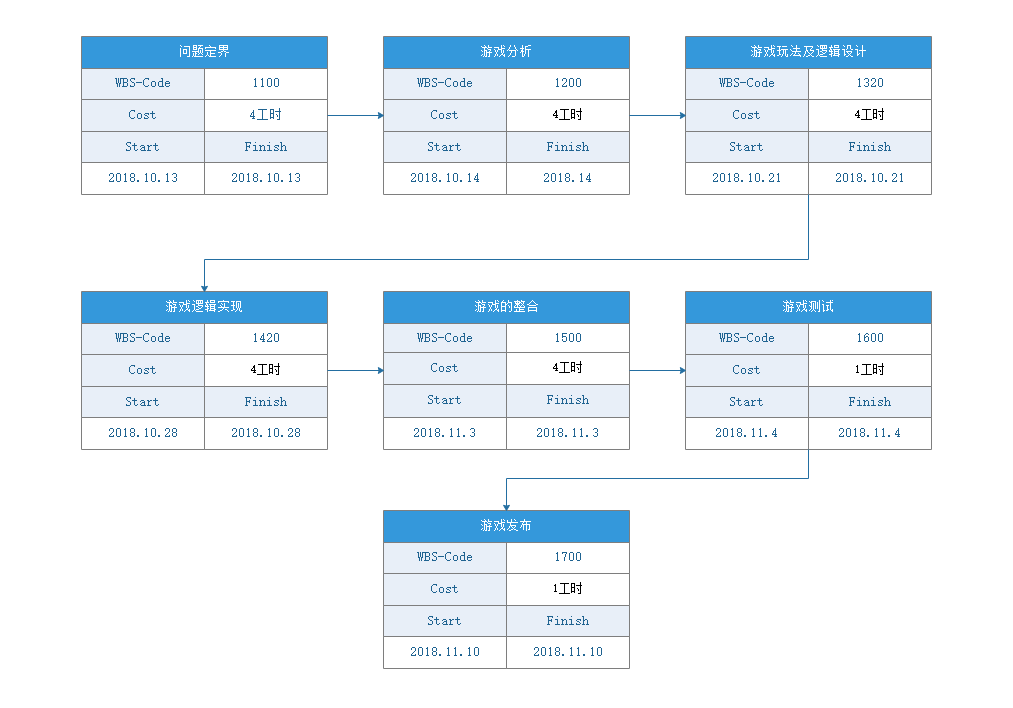
每项的评分依次为：3,0,0,3,5,0,0,0,2,2,2,0,4,5，总和为26。

根据公式得FP功能点为28.21。

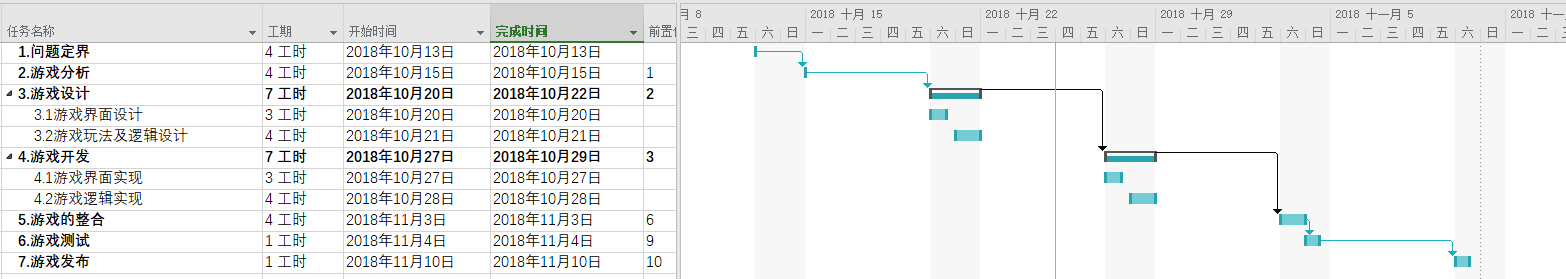
1. **进度管理**
2. 网络图：



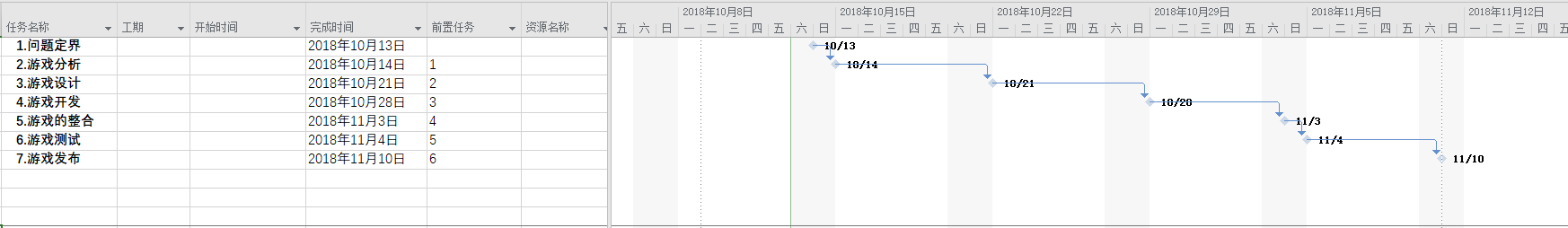
1. 关键路径：



1. 甘特图



1. 里程碑图



1. **风险管理**

项目风险评估表如表4.1所示。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 风险项 | 风险类型 | 风险描述 | 影响程度 | 可能性 | 风险量级 | 优先级 | 风险控制措施 |
| 1 | 人员 | 技术风险 | 开发人员对于开发所需要的技术了解不够充分，需长时间进行学习。 | 高 | 高 | 高 | 高 | 多阅读相关书籍以及上网查阅相关资料，有问题可以及时与有能力的同学求助。 |
| 2 | 开发界面 | 进度风险 | 开发的游戏未达到所预计的样子，游戏模式及界面不符合用户需求。 | 中 | 中 | 中 | 低 | 在开发时及时与客户沟通，并进行软件的迭代。 |
| 3 | 计划编制 | 进度风险 | 计划中有一些不超出个人能力的，不实际的安排 | 中 | 中 | 高 | 中 | 及时根据实际情况修正计划 |

1. **质量管理**
2. 质量控制（QC）
3. 在软件开发过程当中，严格按照软件的生命周期来进行设计软件，使用技术的手段来保证设计的软件符合客户的需求。
4. 在软件开发的过程中，每个阶段的结束都组织人员对开发的软件进行测试评审，对质量进行评价。
5. 严格规范化所有文档的编写，对项目每个阶段所要提交的工作产品制定明确的标准来保证其质量。
6. 对于软件的修改和变更，进行严格控制，每次修改需要进行评估、调度及跟踪的过程。
7. 建立软件质量的度量标准和验证手段。
8. 质量保证（QA）

（1）做好软件开发中规范的执行情况，尤其是违反规范的行为情况。

（2）对于发现的违规情况进行整改，确保不影响软件的质量。

（3）在每个软件活动的后期请同班同学进行正式的评审，来保证软件的质量。

1. **配置管理**
2. 配置项识别：所有配置项都按照相关的规定统一编号，按照相应模板生成，这些配置项都以一定的目录结构保存在配置库中。
3. 配置空间管理：对配置空间按照需求进行划分。
4. 版本控制：在开发过程中，对软件的版本进行控制，每次的版本都提交到GitHub上，同时对系统的不同版本都要进行跟踪和识别。
5. 变更控制：结合人的规程和自动化的工具进行严格的变更控制，严禁随意改动。
6. 配置审计：通过正式技术评审或软件配置审计来保证每次变更的有效性。
7. 状态报告：对于在开发过程中发生的事情，都进行记录与保存。