# 开发计划

|  |  |  |  |
| --- | --- | --- | --- |
| 当前版本 | 修改概述 | 修改人 | 修改日期 |
| 1.0 V | 指定毕设开发计划 | 余国义 | 2021.12.19 |
|  |  |  |  |
|  |  |  |  |

## 部署开发环境

### TOLua框架的部署

<https://blog.csdn.net/linxinfa/article/details/88246345>

### PB协议的使用以及部署

* google protobuf

<https://blog.csdn.net/mao834099514/article/details/53906749/>

* Lua-protobuf

https://zhuanlan.zhihu.com/p/26014103

### PB协议与数据库字段的对应关系

目的：根据在.proto文件中定义的字段，创建对应的表结构。

### 服务器前后端底层通讯功能的搭建

服务器类型：

1. login服务器
2. map服务器
3. talking服务器
4. game服务器
5. Pb服务器

* LoginServer

功能：

1. 接受客户端的连接
2. 管理客户端的连接
3. 对于客户端的登陆判断。

* MapServer

功能：

1. 当客户端登陆成功后，loginServer服务器会将MapServer监听的端口以及ip告诉客户端，通知客户端与MapServer建立连接。
2. 管理客户端链接。
3. 与客户端进行消息通讯。

* TalkingServer

功能：

1. 当客户端登陆成功后，loginServer服务器会将TalkingServer监听的端口以及ip告诉客户端，通知客户端与TalkingServer建立连接。
2. 管理客户端的连接
3. 接受客户端的聊天信息，并将这个聊天数据定时通知PBServer进行保存。将聊天信息进行转发。

* GameServer

功能：

1. 当在MapServer服务器中接受到开始游戏请求后，将GameServer的IP与Port发送给开始对局的客户端，并让他们建立连接，等该所有人准备完毕，然后开始向客户端发送游戏开始标识，以及随机种子，表示游戏开始，并进入第二阶段，等待接受客户端的控制指令。并将每一局的操作指令都发送给PBServer，让其保存，以便于对比赛的回放。
2. 接受客户端链接
3. 管理客户端连接

* PBServer

功能：

1. 分别与GameServer、TalkingServer、MapServer、LoginServer进行链接，接受他们传递的数据，并保存。
2. 接受连接。
3. 管理连接。

### 为实现帧同步而使用的定点数库

因为采用的同步方式是帧同步方式，而这种方式是将计算放在各自的客户端，服务器只需要在开始的时候，向所有客户端发送一个随机种子，然后客户端根据这个随机种子开始各自的计算，在游戏期间，服务器会每隔一个时间帧来接受客户端的操作指令，服务器需要接受到所有客户端的操作指令，再将指令+当前帧数分发给所有客户端，客户端根据这个信息来计算出最新的数据。 在这个过程中，由于所有的计算都是有各自的客户端自行计算的，所以如果出现浮点数计算，因为计算机的精确度是有限的，如果超出计算机的显示区域，这样在进行计算时，会产生误差，并且由于玩家的计算机的精确度也可能不同，所以这样可能会导致，同一个指令在不同的客户端计算的结果可能不同，所以我们需要采用某种方式来避免这种情况。大概有以下几种方法：

1. 使用定点数来代替浮点数计算，即使用定点数库。
2. 使用放大法，将浮点数进行放大一定的倍数并取整，这样虽然丧失了一定的精确度但是也能解决上面所说的问题。

经过深思熟虑，决定使用定点数库来保证客户端的同步。

<https://github.com/aaa719717747/TrueSyncExample>

因为使用放大法，在计算时可能会由于数值过大，导致溢出，所以这也是为什么是使用定点数库，因为成熟的定点数库它对于溢出做了一些特殊处理。

在使用三角函数时，会使用查表法，保证获取的三角函数值是一定定点数。

### 根据需求开发具体功能

ClientEventManager功能。、 注册事件，ClientEventType，立即播放事件，延时播放，

UIManager功能，开启页面，关闭页面，执行可回退的关闭页面。