（这个文档是为了补充之前的《第二轮sprint及后续小会》，关于二轮过后各部分分工）

在一轮过后，我们的棋盘得以展现出来，在二轮Sprint上如何改进就成为了下一步的重点。

**情景假设**

**情景A：** 同学A和同学B想要进行一次对战，需要通过鼠标点击棋子下棋，但是不一定会点在两条线的交点，而是一个以交点为圆心，适当长度为半径的圆内。

如果点在另一个棋的范围内，同色则视为重新选择棋子。而这又提出一个新问题：如何判断现在选择的棋子？ 解决方案：选定的棋子周围出现选择框。

但是如果出现违背规则的走法（将的下一步在九宫格外），程序应该不允许这样的走法实现，及规则的具体实现。

**情景B：** 同学C想要根据自己的棋谱来摆象棋残局。

如果是自己摆谱，就不需要系统自动摆好谱，最好可以在棋盘展现前就可以告诉程序时要用哪个模式，于是就需要一个模式选择。开始摆谱时，需要选择不同的棋子，而不同的棋子出现的次数有上限，如果可以直接点取棋子再点击棋盘摆放则是最方便的做法。而在摆好自己需要的期盼后需要告诉程序我要开始下棋，在这之后则与正常模式相同。

**情景C：** 同学D想记录下这盘和同学E的棋局进行保存，并可以在棋局中看到之前的几步，以后也可以通过读文件的形式进行复盘。

于是就需要一个记录。这个记录应该可以将将每一帧棋局保存下来。考虑到棋谱的通用性以及程序运行的速度，应该以一种规范而简洁的形式记录，而字符串数组（FEN）就是一个不错的选择。读谱时棋盘会在适当的速度下将每一步展现出来，这就需要一个函数，逐行读入记录文件，并更新画面。

在和E下棋的过程中，D发现自己又一步下错了，经过E的同意，进行了悔棋。

以上三个情景应该是最有可能出现的场景，而关于悔棋又可以拓展出各种可能，这一点在之前的《第二轮sprint及后续小会》中有着较为详细的记叙，在此不再赘述。除了场景（故事）外，还有在程序编写过程中必要却不能体现在故事里的过程，这些过程也有专门的分工，具体分工如下。而分工中相同部分的工作为两位同学各自编写，再经由讨论来进行取舍。

**具体分工**

陈晨：自由模式和正常模式下棋子的摆放（重点在自由模式，包括选棋界面及具体摆棋），以及行走（规则的具体应用）

时间：三周

肖佳伟：负责record 功能，FEN和矩阵的相互转换（即棋盘记录，字符串组和矩阵之间转换） 展示功能 （把已有的纪录 逐步像ppt一样演示走法）

时间：二到三周

崔泓睿：将二轮前代码重构为面向对象（类封装），风格统一；测试代码的编写

时间：三周（包含大量学习的用时）

罗吉年：悔棋

时间：二到三周

黄毅力：会议内容的总结记录，管理文档编写，模式选择界面的编写

时间：代码时间一天，记录每次会议后一天

翟拙存：模式选择界面的编写，程序打包相关的学习与编写

时间：一到二周

王旭耀：摆棋部分代码编写

时间：一到二周