

BlackJack 产品文档

使用说明

打开 `index.html`，进入游戏初始界面。点击 **Start Game** 按钮，开始游戏。游戏初始发牌两张，可通过点击 **Hit** 按钮继续发牌，通过左侧 **score** 分数来决定是否继续要牌，当 **score** 点数超过 21 点时，游戏失败，显示“You Bust!”信息。要牌过程中随时可以通过点击 **Stand** 按钮结束要牌，进入判断胜负阶段。若庄家超过 21 点，则玩家胜出，显示“Maker Bust!You Win!”提示，否则按照庄家和玩家的点数大小，点数大者胜出。**Restart** 按钮出现，可点击重新开始游戏。

设计思想

游戏发牌采用一种完全随机算法，通过获得当前时间的毫秒数后让其分别与 4 和 13 取余而获得将发牌的花色以及点数，然后通过一个标记数组判断这张牌是否还在牌堆中，若不在则重新获取当前毫秒数生成一张牌，若在则发出该张牌并标记该牌已被发出。

游戏初始各发牌两张，并显示玩家点数和庄家明牌点数。然后庄家先进行游戏。庄家 AI 游戏标准为，若当前点数小于 17 则继续要牌，否则停止要牌。庄家选择完毕后，玩家进行游戏，由于处理速度极快，所以庄家游戏时间可忽略不计。玩家点数小于 21 点时可以进行要牌和停止要牌两种选择，点数大于 21 点时则游戏失败。若玩家选择停止要牌，则庄家明牌，进入判断胜负阶段。庄家爆则玩家胜，否则通过比较二者点数大小，点数大者胜。玩家可在此时选择重新开始，进入新一轮游戏。

技术实现

游戏逻辑主体由 JavaScript 完成，仅在设置页面大小和显示庄家暗牌两处语句使用了 jQuery。考虑到玩家和庄家两个对象，并且出于高内聚低耦合的标准，特采用了单例模式的设计思想来实现，将玩家分数等设为私有变量，从而一定程度上降低了数据操作的耦合度。