**小学期CPP课报告之App part**

课题：篮球数据可视化应用（俱乐部数据以及球员个人数据的收集与可视化分析）

Github：<https://github.com/lincandong/Basketball_Data_Visualization>

需求分析详见git主页的‘README.MD’

团队分工：

view model与app：林灿东 - 计科1601 – 3160103968

view：梁家宁 - 计科1601 – 3160105677

model：谭泽源 – 计科1601 - 3160103832

**课题要求：**

1.以MVVM模式完成一个自选主题的应用开发

2.使用cpp系列工具链进行开发

**模块功能阐述：**

本模块负责将view，model，view model三个模块综合起来：初始化各个模块、将通知发送器与接收器绑定、将view中的数据与view model绑定、将view中的命令与view model中的命令绑定。

由于mvvm模式下各模块均不耦合，因此绑定都是通过各模块提供的接口实现的。

**源码解读：**

int main(int argc, char \*argv[])

{

//初始化QT模块

QApplication app(argc, argv);

//初始化三个模块（vm中内置model）

view v;

viewModel vm;

//绑定属性改变通知

vm.m.InitSender(vm.getModelReceiver());

//绑定通知发送器与接收器

vm.setScc(v.getCommandReceiver());

vm.setSpc(v.getPropertyReceiver());

//绑定数据

v.setPlayer(vm.getPlayer());

v.setPlayerRank(vm.getPlayerRank());

v.setTeam(vm.getTeam());

v.setTeamRank(vm.getTeamRank());

//绑定命令

v.setPlayerDataCommand(vm.getPlayerDataCmd());

v.setPlayerRankCommand(vm.getPlayerRankCmd());

v.setTeamDataCommand(vm.getTeamDataCmd());

v.setTeamRankCommand(vm.getTeamRankCmd());

//显示界面

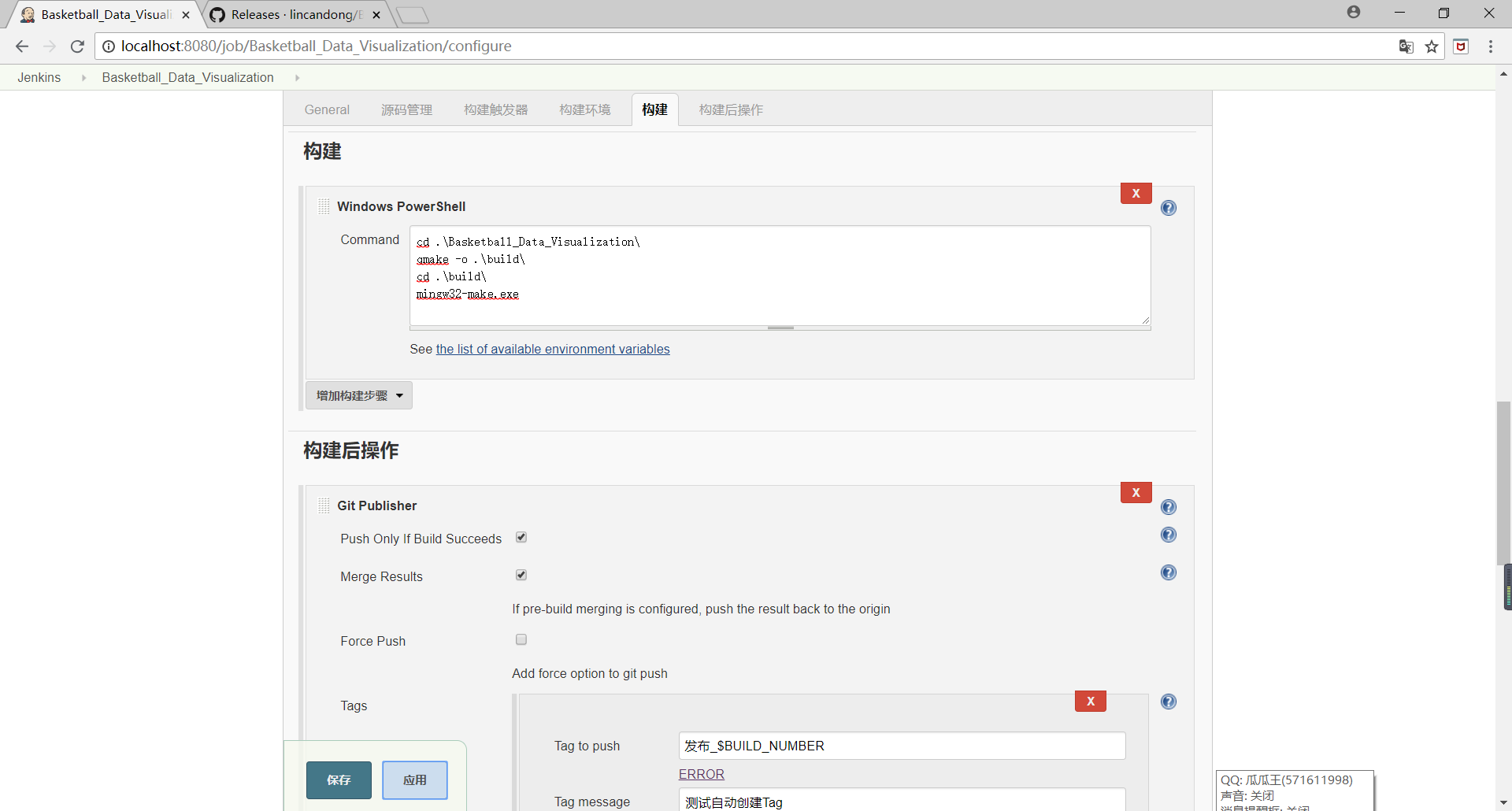
v.show();

return app.exec();

}

**工具链使用：**

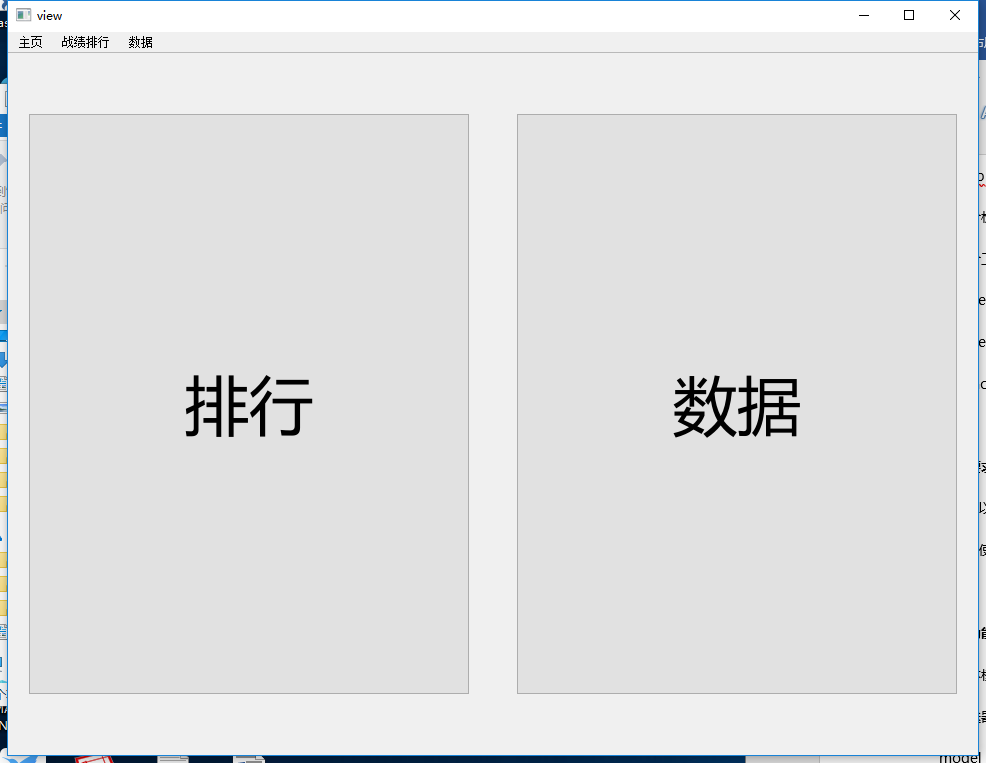
Jenkins作为一个持续集成工具，提供了许多功能插件，我们通过这些功能插件和终端插件在Jenkins上集成自动构建、测试、发布项目的功能，以实现自动部署持续集成，同时将项目成果存储在github上，方便团队协作，并使用git作为版本控制工具以方便开发。



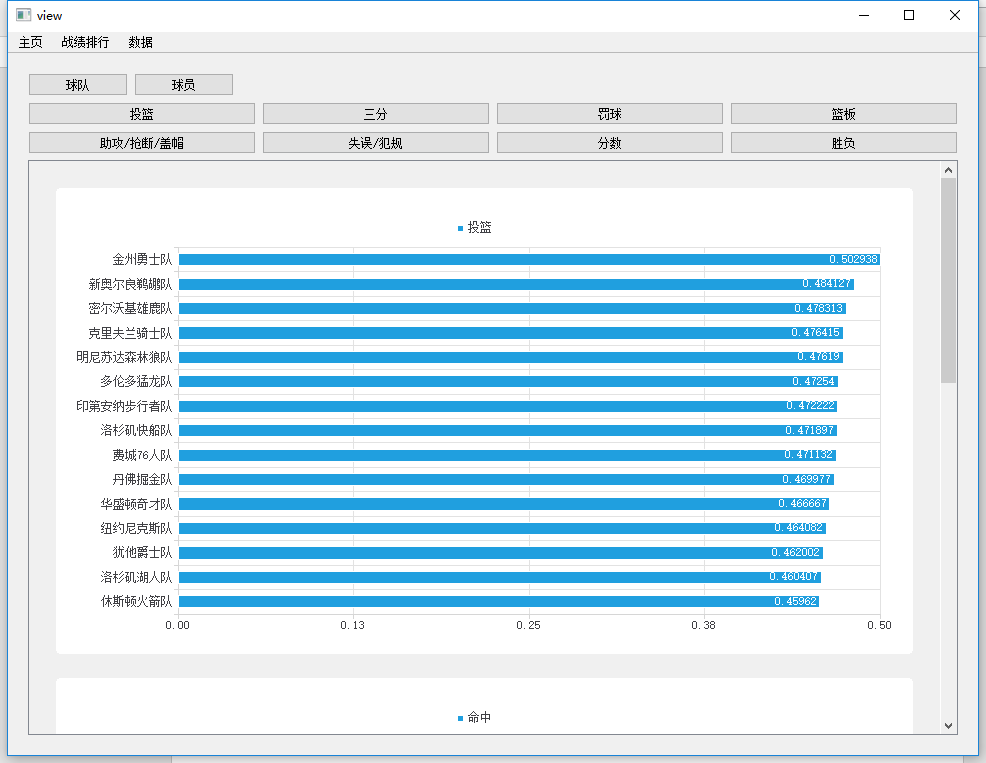
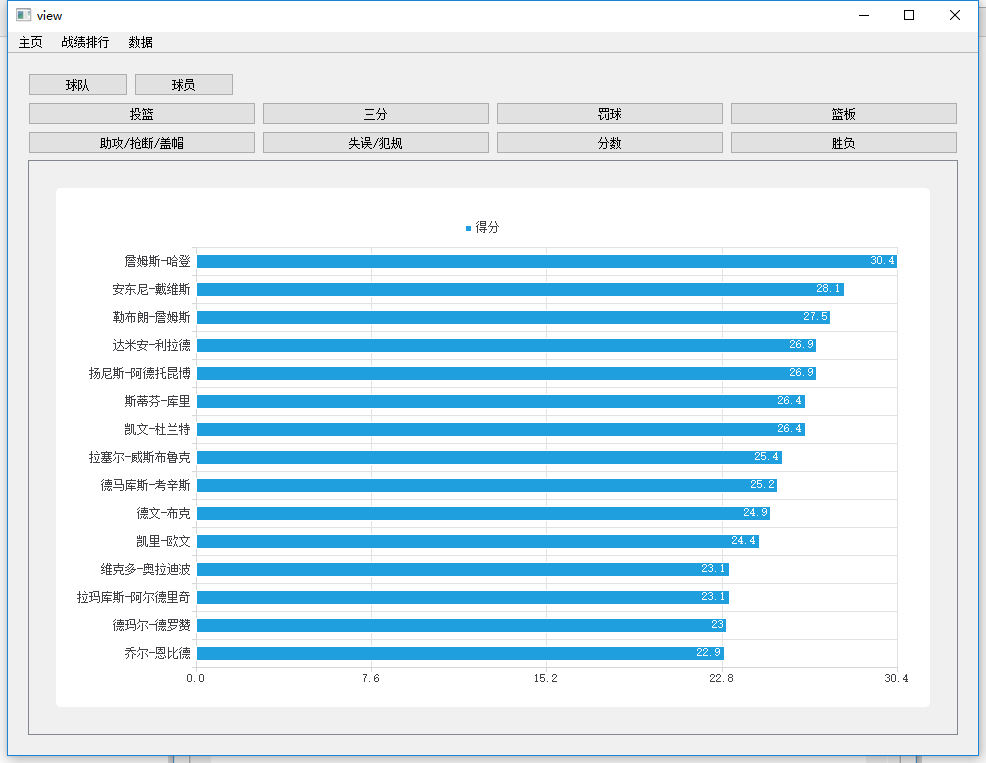
利用Jenkins自动构建并发布在github上。

**运行效果展示：**

程序开始运行时，在主页可以选择查看排行信息或者球员/球队的数据。



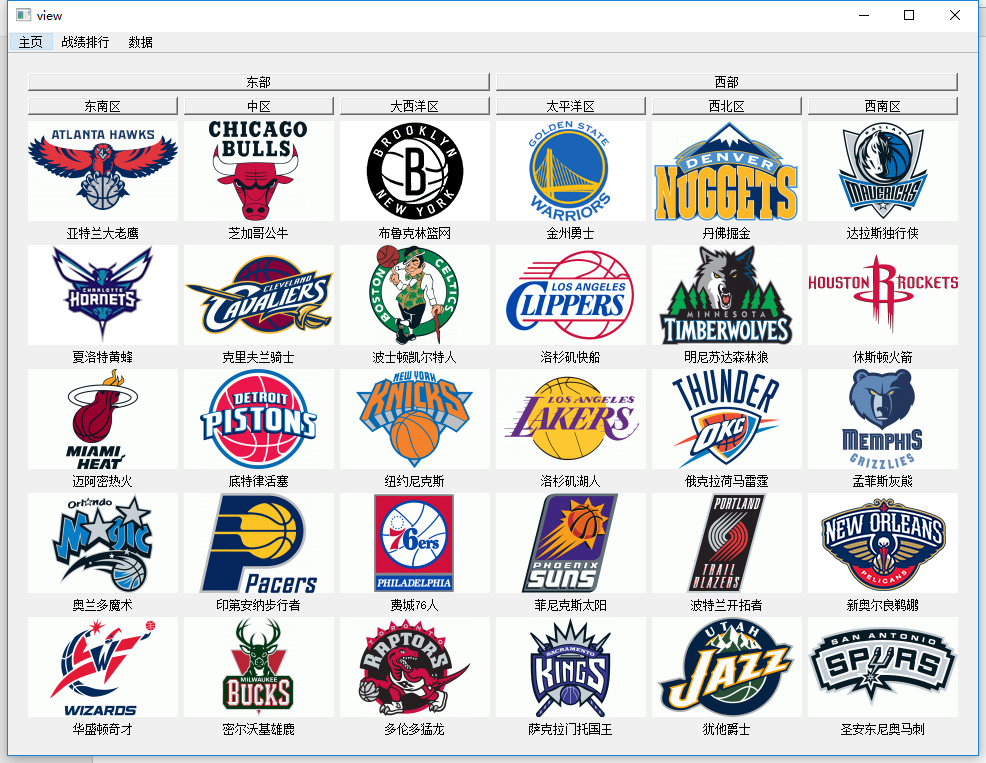
点击排行：

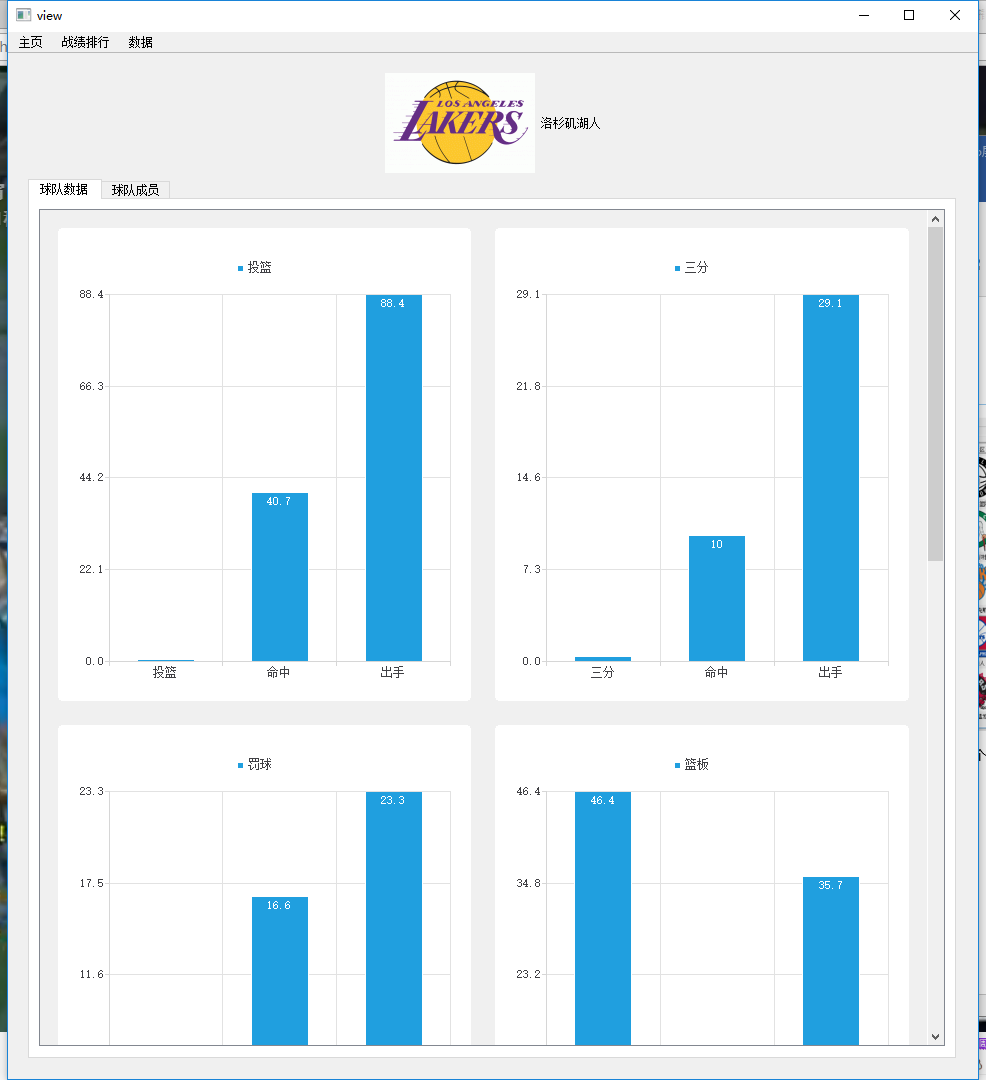
可以在筛选器中选择查看球员还是球队的排行，并且可以查看不同单项数据的排行。

每点击筛选器中的选项，相当于改变view模块向view model发送的命令的参数，进而更新view model中与view绑定的排名数据，进而改变图形界面的显示情况。

点击数据：

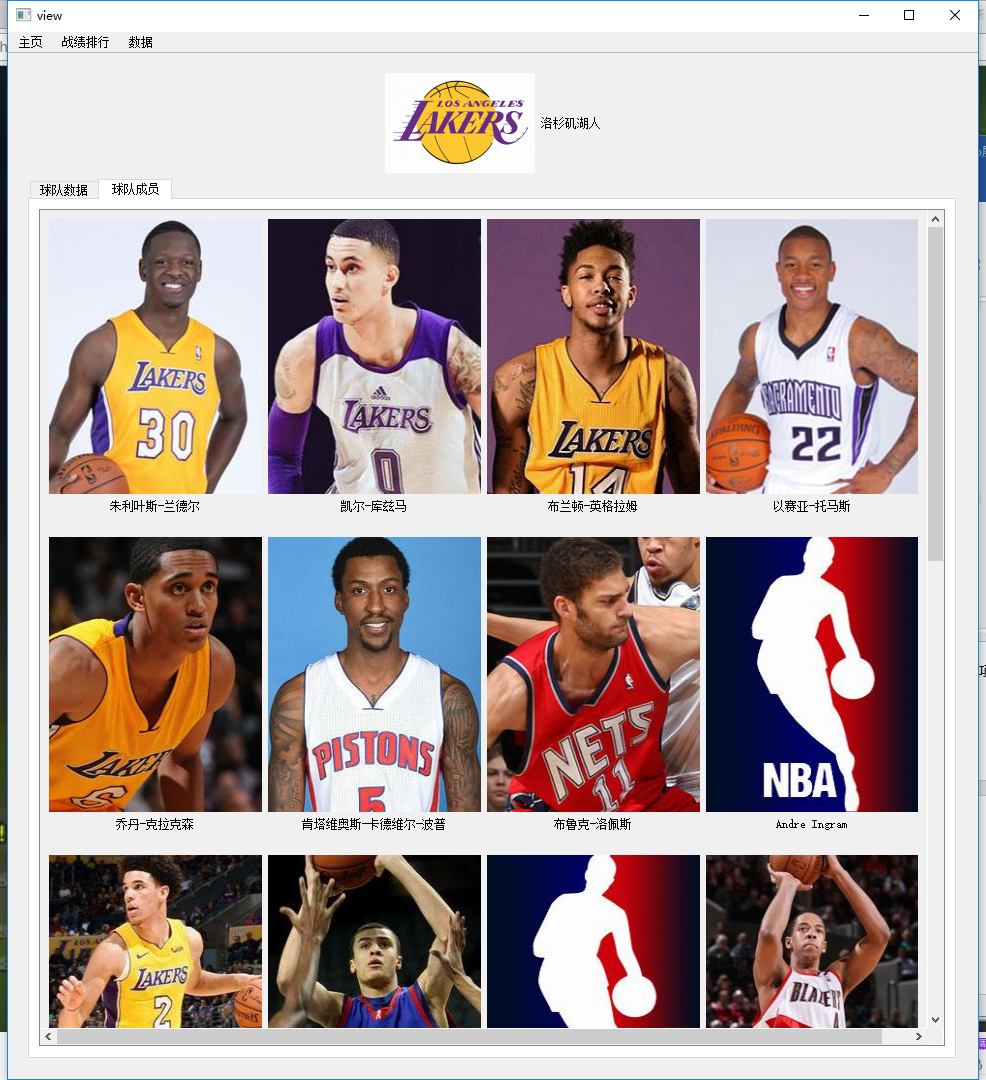


view从资源文件中加载35个球队的名字与图片数据，点击任意球队图标进入后续界面：

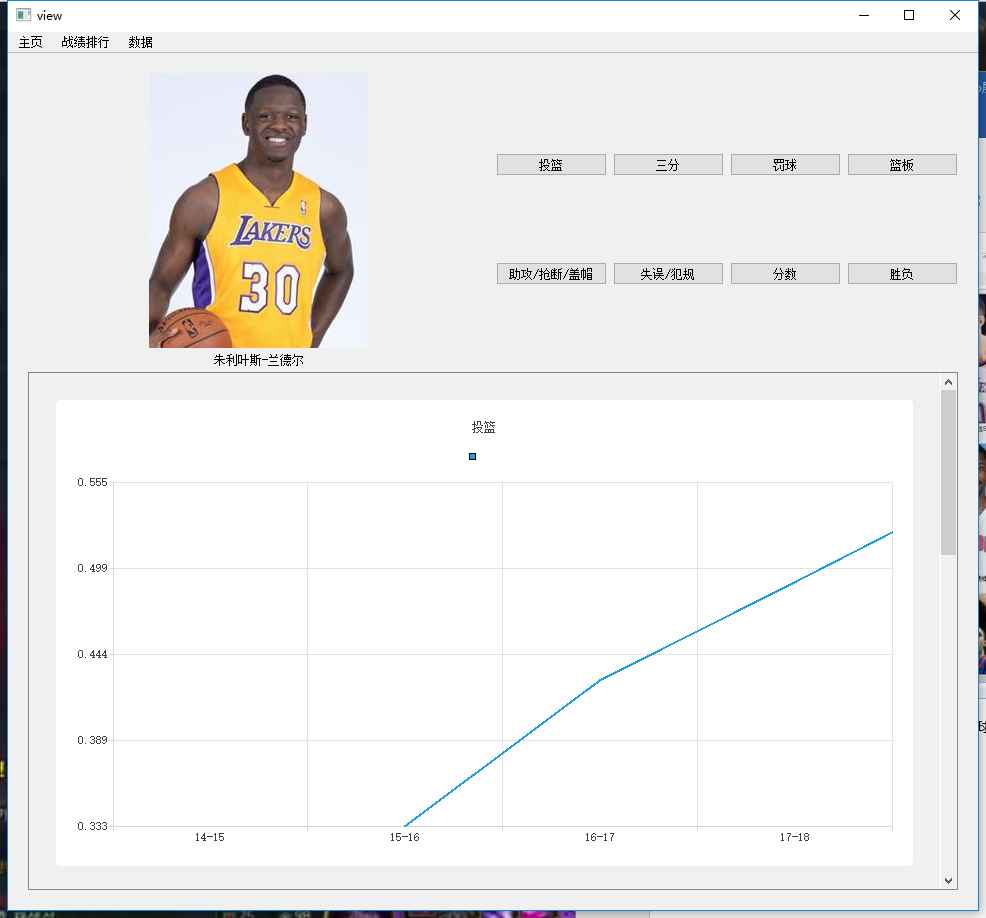


以柱状图的形式呈现球队的各项数据。

点击球队成员选项：



View发送命令，model返回球员信息与图片。点击任一球员图片：



以折线图形式反映球员各项数据。同样的，点击筛选器选项，view发送命令来获取显示图表需要的数据。