# Tiki-Taka Visual项目开发报告

## 1. 引言

**1.1 项目目的**

足球联赛中，球员个人数据、俱乐部联赛数据在众多网站和APP中都是以列表形式存在，统计手段的增强导致每场比赛，每个赛季都可以获取的数据大大增加，传统的列表形式无法两哈哦的呈现这些数据的相关性，而一些第三方的可视化视图实验性大于实用性质，并没有应对球迷的需求，存在统计面没有切合球迷感兴趣的方面的问题。因此，使用数据可视化的方法来让相关的数据呈现出一定相关性，有了一定的使用前景。



图 1传统的足球数据查询平台

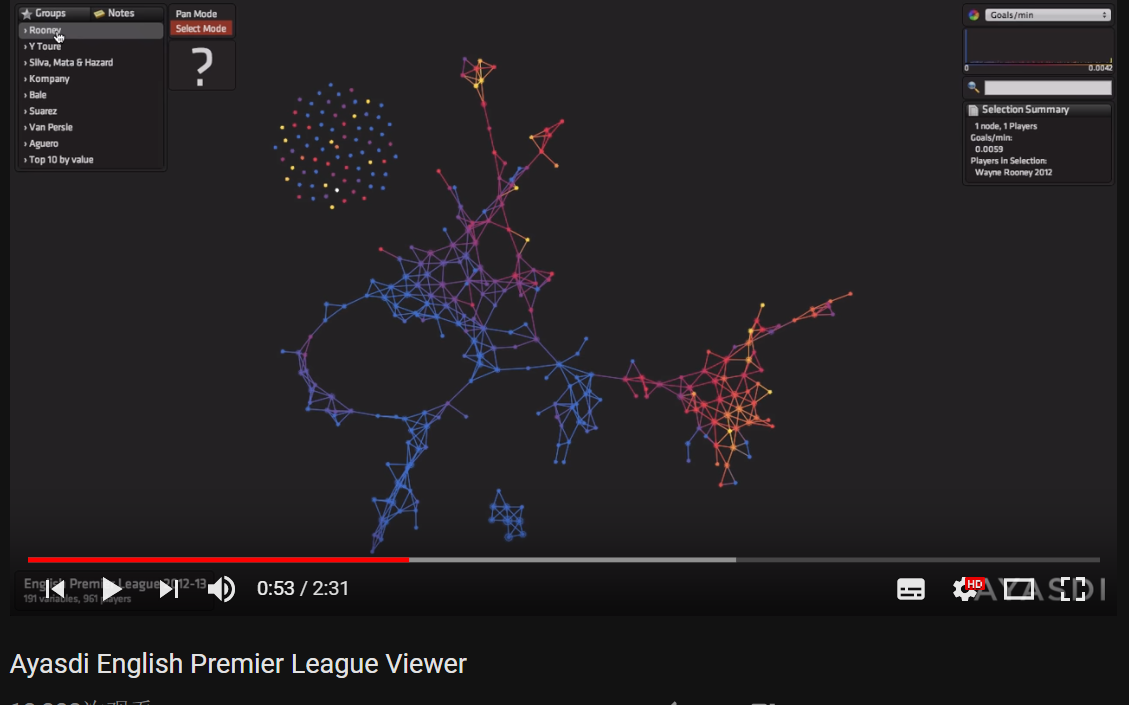


图 2足球相关数据数据可视化

开发足球联赛数据可视化分析的平台，能让球迷方便的查询往届赛事的联赛数据，并且在可视化视图分析之下，清晰的看出球队在球员交易、攻防数据、联赛成绩上的变化，并与视图进行互动，使得本项目能比传统足球联赛数据展示方式更具优势。

**1.2 背景**

|  |  |
| --- | --- |
| 项目名称 | Tiki-Taka Visual |
| 用户群体 | 足球爱好者、业余球员教练 |
| 功能 | 查询欧洲五大联赛往届联赛数据，并在条件允许的情况下查询本季联赛的数据，并将这些数据进行可视化展示。 |

**1.3 参考资料**

**Python 《Beginning python 3》**

**D3数据可视化开源库** [**https://d3js.org**](https://d3js.org)

**Pyqt 5.0 技术文档**

**《数据库系统概念第六版》**

**Scrapy** [**https://github.com/scrapy/scrapy**](https://github.com/scrapy/scrapy)

**Echarts** [**http://echarts.baidu.com/feature.html**](http://echarts.baidu.com/feature.html)

## 2. 项目概述

**2.1 工作内容**

|  |  |
| --- | --- |
| 内容 | 描述 |
| 数据收集 | 搜集往届五大联赛俱乐部、赛事各项指标、球员的数据库 |
| 市场调研 | 调研比对软件或者平台的优劣点 |
| 开发文档 | 开发项目流程文档 |
| 项目开发 | 代码实现 |
| 项目测试 | 测试平台稳定性并作出bug修改 |
| 支持文档编写 | 用户使用支持文档编写 |

**2.2 主要参加人员**

|  |  |  |
| --- | --- | --- |
| 组员 | 学号 | 职责 |
| 郭智溢 | 2018226045016 | 文档、前端、后端 |

## 3. 实施计划

**3.1 工作任务的分解与人员分工**

|  |  |  |  |
| --- | --- | --- | --- |
| 任务 | 负责人 | 需要具备的知识 | 开始时间 |
| PyQT客户端前端 | 郭智溢 | Python、PyQT | 第9周 |
| 数据库接口开发 | 郭智溢 | Python、Sqlite3 | 第10周 |
| 制定前后端数据接口 | 郭智溢 | Python、javascript | 第10周 |
| 平台测试 | 郭智溢 |  | 第14周 |
| 用户手册编写 | 郭智溢 |  | 第15周 |

**3.2 项目的生命周期模型采用敏捷开发模型**

在制定开发Tiki-Taka Visual的计划后，将对需要这类应用的球迷进行需求分析，并完成设计文档和程序实现，最后在产品可以正常运行后进行软件测试和运行维护。

**3.3 进度安排表**

|  |  |  |  |
| --- | --- | --- | --- |
| 软件开发计划 | 起始时间 | 结束时间 | 综述 |
| 开发文档 | 第八周 | 第八周 | 完成开发文档 |
| 需求分析 | 第九周 | 第九周 | 完成项目需求分析 |
| 技术实现评估 | 第九周 | 第九周 | 对python后台和html前台联动进行评估 |
| 概要设计和详细设计文档 | 第九周 | 第十周 | 完成软件页面的细节设定 |
| 开发 | 第十周 | 第十四周 | 开发过程 |
| 测试 | 第十四周 | 第十四周 | 测试软件以及修改bug |
| 使用指南 | 第十五周 | 第十六周 | 用户使用指南 |

**3.4** **任务优先级**

首先选择合适的视图，对开源库提供的视图如何嵌入PyQT界面中进行调研，过后使用python来对前后端进行开发，部分数据处理源自python后台。

**3.5 开发者能力评估**

目前开发者对于js、python、sql都有一定的知识储备，但是对于PyQT的新组件Webengine的知识是比较欠缺的，这一块涉及到如何。

**3.6 风险管控**

|  |  |
| --- | --- |
| 风险 | 备选 |
| Python后端数据库与前端逻辑的联动出现技术问题。 | 直接使用python来完成独立的客户端 |
| 客户端没法做到多视图的联动。 | 为部分视图添加单独的界面 |

|  |  |
| --- | --- |
| 管控 | 描述 |
| Git跟踪式开发 | 使用git跟踪进度、bug |

## 4.支持条件

**4.1开发硬件支持**

客户端开发机

|  |  |
| --- | --- |
| cpu | I3 8100 |
| 内存 | 8GB |
| 硬盘 | 256GB(ssd)+128GB(ssd)+1TB(机械硬盘) |
| 显卡 | GTX 1060GB |
| 系统 | Win10 pro |

服务器

同客户端开发机为同一台机器，先期调试为本地调试。

**4.2** **开发软件支持**

|  |  |
| --- | --- |
| Vscode | Python与html开发工具 |
| Pycharm | Python开发工具 |
| Git for windows | 版本控制工具 |

**4.3** **开发工具包支持**

|  |  |
| --- | --- |
| D3.js | 数据可视化开源库 |
| mongodb | 数据库 |
| Python V3.6 | Python环境 |

**4.4数据存储结构**

本地存储，在成本核算可以接受的情况下使用云存储来存储往届赛事数据。