教育部高等学校大学计算机课程教学指导委员会

中国大学生计算机设计大赛



软件开发类作品文档简要要求

作品编号：　　　　　　　 67982

作品名称：\_**基于B/S+C/S混合部署架构的分布式射箭比赛管理系统**

作　　者：　 吴烜圣、郭菁菁、杨非池

版本编号：　　 　　 V3.0.1

填写日期：　　　　　　 2019-5-29

填写说明：

1. 本文档适用于**所有**涉及软件开发的作品，包括：软件应用与开发、大数据、人工智能、物联网应用；
2. 正文一律用五号宋体，一级标题为二号黑体，其他级别标题如有需要，可根据需要设置；
3. 本文档为简要文档，不宜长篇大论简明扼要为上；
4. 提交文档时，以PDF格式提交本文档；
5. 本文档内容是正式参赛内容组成部分，务必真实填写。如不属实，将导致奖项等级降低甚至终止本作品参加比赛。

目录

[第一章 需求分析 3](#_Toc31828)

[第二章 概要设计 4](#_Toc6911)

[第三章 详细设计 6](#_Toc3512)

[3.1关键技术：B/S和C/S混合部署架构 6](#_Toc29975)

[3.2 数据结构设计 6](#_Toc15689)

[3.2.1 持久化数据 7](#_Toc29425)

[3.2.2 临时数据 8](#_Toc19376)

[3.2.3 缓存数据 8](#_Toc9987)

[3.3 客户端界面设计 9](#_Toc11334)

[3.3.1 界面合理性 9](#_Toc16121)

[3.3.2 界面一致性 10](#_Toc9951)

[3.3.3 系统操作 10](#_Toc14560)

[3.3.4 系统帮助 11](#_Toc26802)

[3.3.5 系统响应 11](#_Toc7918)

[3.4 网页端界面设计 11](#_Toc1424)

[3.4.1简洁的设计元素 11](#_Toc26205)

[3.4.2 简单的交互设计 12](#_Toc13260)

[第四章 测试报告 13](#_Toc12840)

[4.1测试说明 13](#_Toc8331)

[4.2 测试环境及时间安排 13](#_Toc175)

[4.3 主要修复的测试项目 13](#_Toc26442)

[4.4 测试结果 14](#_Toc17298)

[4.5 本项目应用成果 14](#_Toc20190)

[第五章 安装及使用 15](#_Toc6975)

[第六章 项目总结 16](#_Toc16192)

# 需求分析

射箭是奥运会正式比赛项目，也是传承中国传统文化的重要载体。一场射箭比赛包含报名、成绩汇总、靶位编排、数据统计等需要信息系统进行有效管理的环节。“Excel统计，手动编排”的方式效率低、错误率高，遏制赛事规模的扩大也遏制了中国文化的推广。

目前，国内外射箭比赛管理系统有两类：“单节点架构”系统和“B/S 架构”系统。单节点系统由一人将所有纸质版成绩依次录入电脑，但单人短时间处理大量数据不仅错误率高且效率低。B/S架构系统为了应对大型赛事由多名志愿者在赛场用手机进行成绩录入，将数据直接存入云端。这种方式不仅不符合锦标赛的要求，且过分依赖网络，网络中一但出现问题就会影响赛事顺利举办。这些解决方案的出现虽然在一定程度上促进了赛事规模的发展，但因技术方案本身的局限性，仍存在着稳定性较差、效率低下的不足。

为提高对赛事的管理效率，保证赛事的顺利进行，重新设计一款赛事管理系统显得尤为重要。赛事组委会对新系统提出了：自动编排每轮次比赛的靶位、多终端进行成绩录入、方便运动员了解实时成绩、赛后导出成绩册、在线报名、在线客服等功能要求。

综合上述架构的优缺点，我们提出一款基于PC客户端并可拓展局域网多节点服务和Web服务的分布式射箭比赛管理系统(Distributed Archery Events Supporting System)作为解决方案。该系统的英文简称为DAESS，中文别名为“赛事管家”。新系统从报名环节开始，对赛事每一个环节进行信息化统一管理，能高效、稳定地满足组委会的所有需求。DAESS解决方案能够覆盖以下9个功能特性：

1. 靶位自动编排：自动安排每一轮次每位运动员的靶位号；
2. 非强依赖网络：各终端的链接不依赖互联网，网络不稳定不会导致系统瘫痪；
3. 可支持锦标赛：系统中的电子版成绩必须有纸质版签名成绩单作为凭证；
4. 导出成绩报表：所有比赛结束后一键生成所有比赛记录的汇总成绩册；
5. 并行成绩录入：同一时刻有多个终端进行成绩录入操作；
6. 成绩实时查询：运动员或教练员可以通过网络立刻查询到刚刚一场比赛的成绩；
7. 在线自主报名：组委会通过网络规范化地采集有意愿参赛人员的相关信息；
8. 数据主从备份：当核心节点由于意外瘫痪时，能立刻启用备份数据节点；
9. 智能答疑客服：基于NLP技术智能在线解答常见问题，以减少志愿者的工作。

DAESS和单节点架构系统、B/S架构系统的功能性比较见表1。

表 1竞品对比分析表

|  |  |  |  |
| --- | --- | --- | --- |
|  | **单节点系统** | **B/S架构系统** | **赛事管家（DAESS）** |
| **靶位自动编排** |  |  |  |
| **可支持锦标赛** |  |  |  |
| **非强依赖网络** |  |  |  |
| **导出成绩报表** |  |  |  |
| **并行成绩录入** |  |  |  |
| **成绩实时查询** |  |  |  |
| **在线自主报名** |  |  |  |
| **数据主从备份** |  |  |  |
| **智能答疑客服** |  |  |  |

# 概要设计

射箭比赛管理系统有两类用户，一是负责使用系统的裁判员、志愿者，另一类是通过系统获取和反馈信息的运动员、教练员。因此，本系统（DAESS)采用C/S模型与B/S模型相结合的方式。需求按照面向的用户群体可分为用Web提供支持的功能（信息公示、在线报名、智能客服、成绩查询）；以及用App支持的功能（靶位编排、成绩录入、编制成绩册）。图1展示了DAESS不同架构群与终端用户的交互过程。

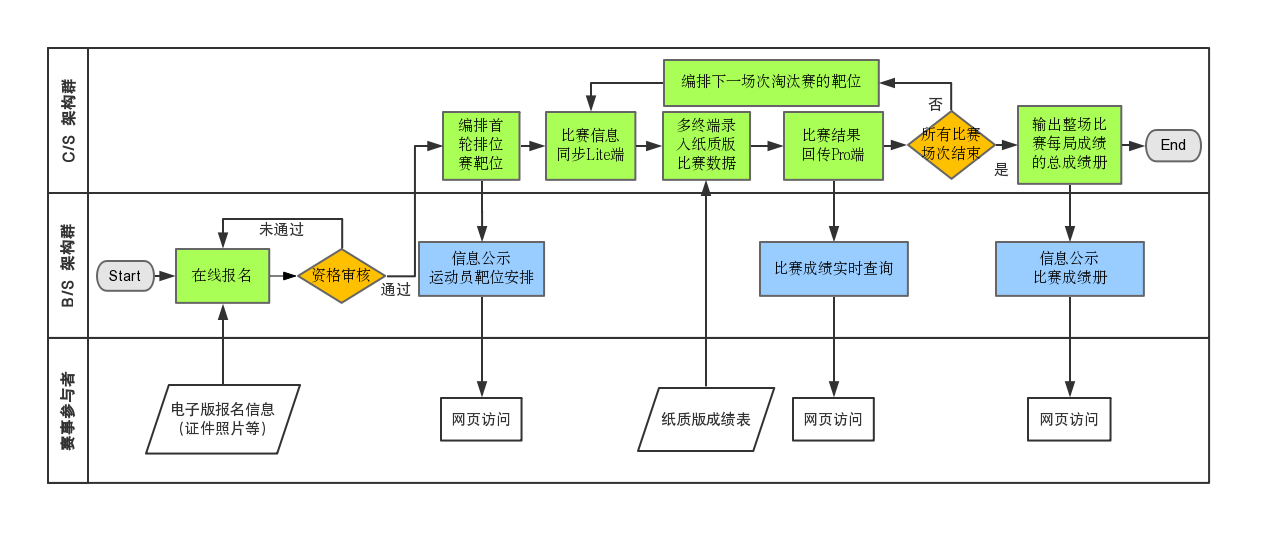


图 1 DAESS用户交互流程图

DAESS的**创新点**在于削弱了各组件间的强依赖性，使得本系统即使在不具有完整B/S和C/S模块时，也可稳定提供部分基本功能——**B/S + C/S混合部署架构。**DAESS通过降低B/S和C/S架构中各模块的耦合性，实现业务逻辑上不同模块之间的灵活组合，使得DAESS成为一款在弱网络环境下同样具备为锦标赛级别赛事提供分布式支持的射箭比赛管理系统。图2描述了本系统B/S+C/S混合部署架构不同模块的层次结构和调用关系。

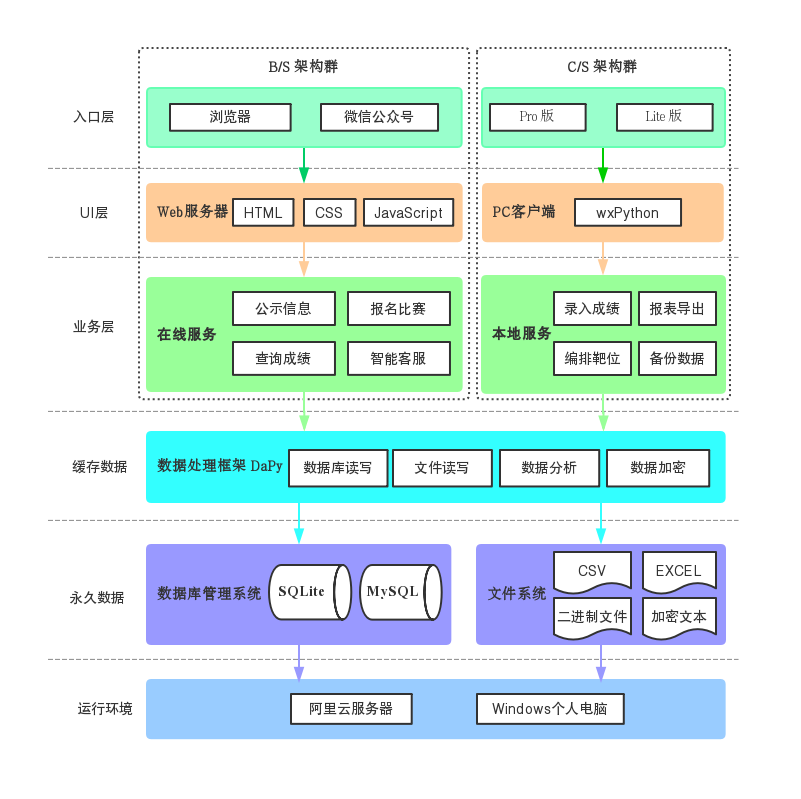


图 2 DAESS逻辑架构图

得益于B/S + C/S混合部署架构，赛事主办方可依据比赛的规模、场地网络环境、志愿者数量以及预算等客观条件将整套系统拆解为：单节点模式、C/S架构、B/S + C/S架构和B/S + C/S联合架构四种形式。图3描绘了本系统B/S + C/S联合架构情况下的物理关系。

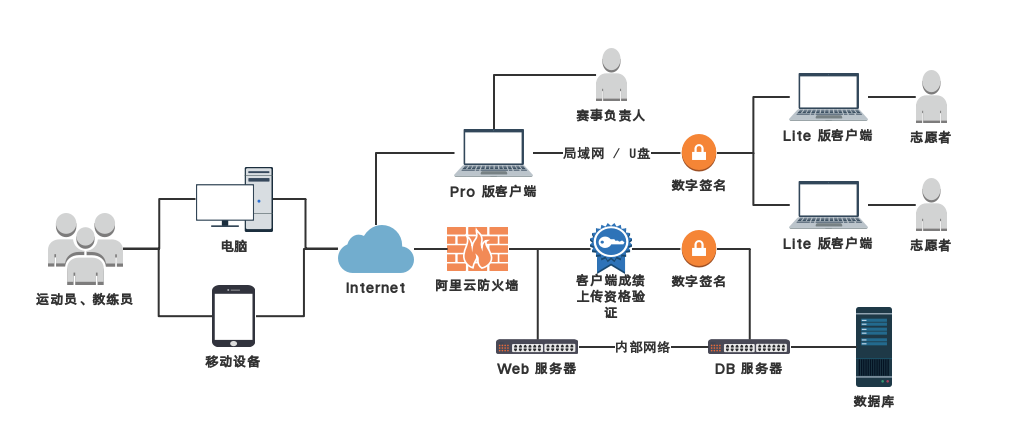


图 3 DAESS物理架构图

在界面设计上，本系统分别采用HTML + CSS技术开发了网页版GUI，wxPython开发了桌面端GUI。考虑到很多工作人员是临时招募的非专业志愿者，为了降低这些半专业用户的培训时间，我们对桌面端特别区分了专业版（Pro）和精简版（Lite）客户端。图4是本系统UI设计过程中不同终端对功能的支持情况。

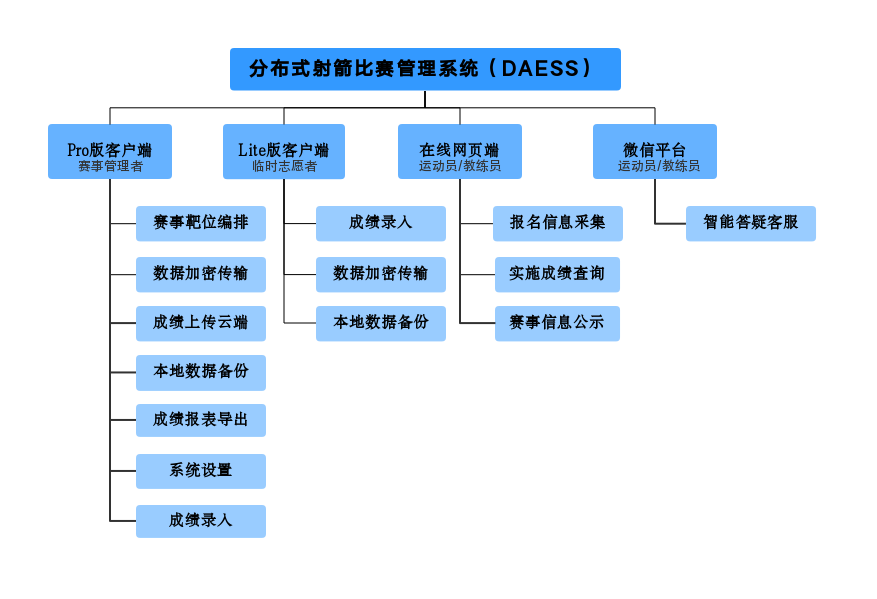


图 4 DAESS功能结构图

在客户端中，主界面的菜单栏由核心功能的按钮共同组成。用户每单击一个按钮后，系统会弹出新的窗口供用户进行操作；同时锁定主页面中其他按钮，防止志愿者或赛事管理人员出现误操作的情况。

在网页端中，运动员可以访问三个页面，分别为首页、报名页面和成绩查询页面。运动员输入网址后进入首页，在首页中包含赛事的基本信息（如：比赛时间、举办场地等）；详细赛事信息可通过点击链接下载完整赛事文件。在首页点击“现在报名”按钮，跳转进入报名录入界面，用户须填写表格完成报名。在首页点击”成绩查询“按钮，进入成绩实时查询界面，用户可通过不同字段进行成绩排序，以及浏览不同场次比赛成绩。

在微信平台中，运动员或教练员只需关注公众号并发送信息，便可得到人工智能客服帮助。例如，领队可向智能客服询问中午的用餐安排，运动员可以询问下一场次的开赛信息等。

# 详细设计

## 3.1关键技术：B/S和C/S混合部署架构

相较于B/S和C/S联合架构，本系统将“子架构群可独立提供部分服务的联合架构”称为**C/S和B/S混合部署架构**。借助B/S + C/S混合部署架构模块化分解各组件的功能，提高了部署的灵活性，并增强了恶劣环境的适应性。图5展示了混合部署架构的搭架过程。

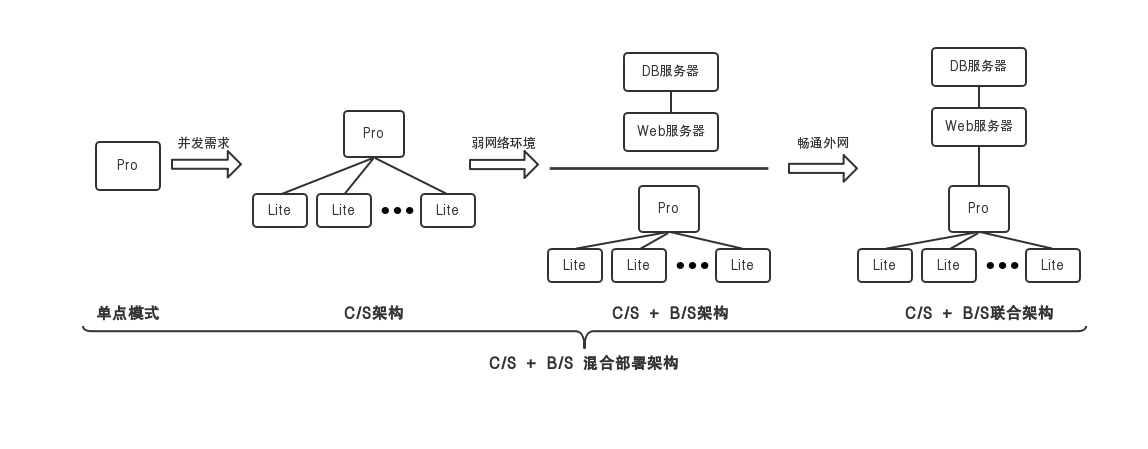


图 5 DAESS混合部署架构

（1）单节点模式：使用一台 Pro客户端以实现编排靶位、成绩录入和成绩报表导出三个最核心的业务功能。由于此时缺乏并发性，仅适用于50人以下的比赛规模。

（2）C/S架构：添加多个仅成绩录入的Lite客户端，Pro和Lite之间通过局域网或者U盘同步数据。增强成绩录入功能的并发性能，使系统轻松应对参赛人数达数百人的大型赛事。相比较于基于B/S模型的商业联赛系统，此架构在网络环境较差时的系统仍然可以工作。

（3）B/S架构：部署Web服务器和DB服务器提供信息公示、在线报名和智能答疑客服三个额外的业务功能，以便于向运动员、教练员群体采集和发布相关的信息。

（4）C/S + B/S联合架构：网络环境较好时，Pro将每轮次的比赛数据上传至DB服务器，运动员和教练员便可以通过Web服务器的页面使用“在线成绩查询”功能。

（5）C/S + B/S混合部署架构：在不同的赛场环境、赛事规模以及运营预算等条件下，用户可灵活选择上述任意一种架构为比赛提供支持。

## 3.2 数据结构设计

本系统中的数据结构按照数据的用途及应用场景区分为“持久化数据”、“临时数据”和“缓存数据”三个数据类型。每一个数据类型在不同场景下都选择了综合考虑最优的数据结构进行有组织地存储，具体的设计细节在以下三个小节中进行了讨论。

表 2 DAESS中的数据结构类型

|  |  |  |  |
| --- | --- | --- | --- |
| **数据类型** | **应用场景** | **使用者** | **数据结构** |
| **持久化数据** | 比赛记录 | Pro、Lite | SQLite3 |
| 参赛报名信息 | DB Server | SQLite3 |
| 导出的成绩报表 | 用户、Pro | Excel |
| 备份数据库 | Pro | 凯撒加密文本文件 |
| **临时数据** | 局域网数据传输 | Pro、Lite | 哈希加密文本文件 |
| 比赛成绩上传云端 | DB Server、Pro | Pickle二进制文件 |
| **缓存数据** | 生成报表、靶位编排 | Pro | DaPy数据框架 |

### **3.2.1 持久化数据**

持久化数据包括系统需要调用但不必加载到内存的SQLite3、MySQL数据库数据，和便于用户跨应用读取的Excel文档类数据两种基本类型；并延伸出了一种用于数据备份的加密文本数据。

（1）数据库存储

常见的关系型数据库主要有SQLite3和MySQL两种。其中，为支持客户端的数据存储，我们选择了具有免安装特性的SQLite3;而针对于部署在云端服务器，我们选择具有高稳定性和高并发能力的MySQL。在项目的运营初期，考虑到实际的需求与成本的限制，在云端服务器上的数据库先以SQLite3作为代替。图6是客户端数据库各个表格之间的E-R图。

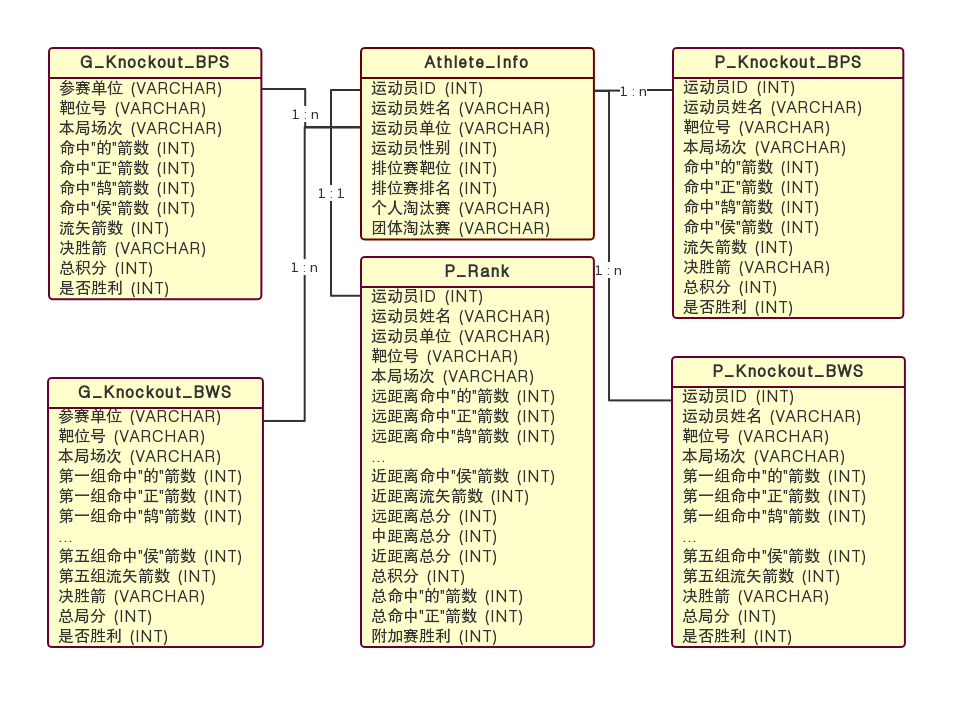


图 6 DAESS客户端数据库表E-R关系图

在客户端数据库中总计有6张表，分别为运动员信息表Athlete\_Info、个人淘汰赛成绩表P\_Rank、个人局胜制淘汰赛表P\_Knockout\_BWS、个人积分制淘汰赛表P\_Knockout\_BPS、团体积分制淘汰赛表G\_Knockout\_BWS和团体局胜制淘汰赛表G\_Knockout\_BPS。Athlete\_Info记录了所有运动员的属性信息，特别生成了唯一运动员识别码（ID）。ID为一个4字节长的INT，前两位代表学校，第三位代表运动员性别，最后一位对应的编号。ID字段解决了在大规模赛事中出现的运动员重名现象导致的违反数据库记录“唯一性”条件的问题。因此，ID作为“生成成绩册”等跨表查询功能最为重要的参数，被记录在了所有客户端数据库的基本信息表中。

（2）文件存储

DAESS主要使用了两种文件格式来存储数据，分别为:报名所需的证件照片(以JPG、BMP、PNG 等图片格式存储)及客户端生成的赛事靶位编排文件或者赛事结果报告文件(以Excel 文件的形式存储)证件照片存储在服务器上，在存储路径下分布着以学校名命名的子文件夹，每一个子文件夹中以报名运动员的姓名命名了该照片。Excel 文件存储在运行目录下的两个子文件夹，命名的规范均为中文报表名称，如“男子个人决赛靶位安排.xls”或“个人排名赛总成绩表.xls”。

（3）加密存储

系统在每次关闭时自动将最新的SQLite3数据库文件以加密的方式存储至本地，其本质仍是数据库结构，但以二进制按照一定密钥进行凯撒加密。这种数据加密存储的方式使得该SQLite3数据库不会被外部的数据库软件恶意打开或访问。

### **3.2.2 临时数据**

临时数据格式被定义为系统间或者模块间用来交换信息用的数据格式。DAESS中，临时数据有两个用途：C/S架构群中的客户端之间的数据同步及客户端与服务器的数据同步。

（1）客户端与客户端数据同步的临时数据：数字签名的数据文件

在数据局域网传输过程中，发送方会将数据的明文写入文件。先将数据明文用HASH加密，将结果拼接密钥，对于拼接的字符串再次用HASH加密,在上述文件的末尾记录上第二轮加密后的值。发送方通过局域网或者U盘分享该数据文件。接收方在读取到数据文件后，先效验该数字签名是否有效，然后再将该数据导入到本地数据库。图7说明了上述的数字签名过程。

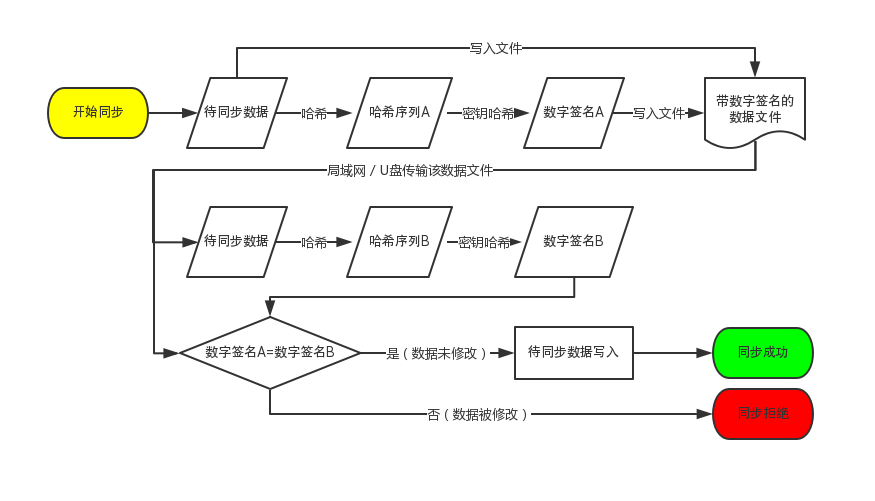


图 7数字签名的加密与效验流程

（2）客户端与服务器数据同步的临时数据：二进制数据文件

在数据客户端上传至数据库服务器的过程中，通过序列化数据的方式将缓存中的数据结构以二进制的形式写入本地文件。将此文件用HTTPS协议中的POST方式上传至DB服务器，通过反序列化加载该数据至缓存，完成数据上传过程。由于HTTPS协议是一种“端到端”的协议，所以此过程中的数据安全性由传输协议保护。另外，生成的临时二进制文件不同于JSON等Human-Readable文件，保证了数据的私密性。



图 8 HTTPS的临时数据共享流程

### **3.2.3 缓存数据**

缓存数据结构被定义为内存中用于数据分析或处理的数据格式。DAESS的缓存数据主要用于报表的编制或编排靶位任务。上述任务需使用多张数据表，经过复杂的逻辑关系处理。传统的SQL语句完成上述问题较为复杂，不利于代码的可读性和维护性，故我们的缓存数据使用Python中的数据处理框架DaPy[[1]](#footnote-0)来完成。DaPy中的SeriesSet数据结构提供了Groupby, Select, Sort等基本的单表函数，也包含Merge、Extend、Join等多表操作功能。

## 3.3 客户端界面设计

图9是DAESS客户端主界面，左侧为功能性菜单栏，右侧是动态背景墙，上部具有页面导航，下部分有提供辅助信息状态栏。



图 9 DAESS客户端版主界面

本系统客户端版坚持以用户体验为中心设计原则，界面直观简洁、操作方便快捷，用户接触软件后对界面上的功能理解充分，下述4个范式用来规范客户端部分的界面设计。

### **3.3.1 界面合理性**

（1）防止用户被过量信息包围；

（2）按照用户习惯布局；

如图10所示为系统淘汰赛阶段的成绩录入界面。在该界面的上方有清晰完整的页面导航栏，帮助用户清晰地认识该界面的主要功能。数据被按照功能合理地展示在了页面的上下部分，帮助记分员快速理清数据的逻辑。各个输入框按照从左至右，从上至下的顺序可以被Tab键依次切换。



图 10淘汰赛成绩录入界面

### **3.3.2 界面一致性**

界面的框架结构、字体、标签风格、颜色等方面确保一致。

1. 使用一致的标记、标准缩写和颜色，同一个功能描述仅使用一个词汇；
2. 字体的颜色均以黑底白字出现，对于不可修改字段，采用灰色文字来显示；
3. 所有的输入操作都设计合法性检验，能够准确地告知用户潜在的错误；

图11是排位赛成绩录入窗口，也是记分员最常用的工作界面。若记分员输入了错误的箭支数将会显示红色的报错，在此情况下本次提交将会被拒绝。另外，左侧的“查询信息”与右侧的“提交记录”两个按钮是互斥开放的，即一个可用时，另一个按钮必然不可用。该规则确保了用户录入的成绩一定是对应于当前运动员信息的。

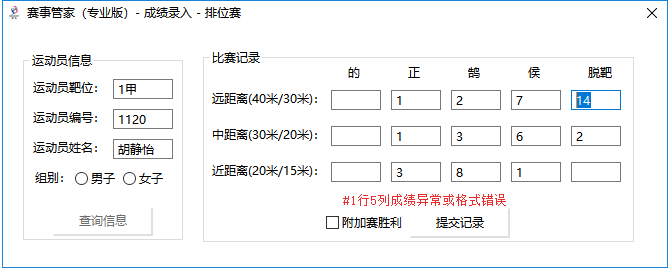


图 11成绩录入功能的输入合法性检验

### **3.3.3 系统操作**

（1）确保用户在仅使用键盘的情况下也可以流畅地完成常用的业务操作；

（2）查询检索类功能，会有提示信息，当用户输入后敲下回车键自动触发查询；

（3）在进行一些不可逆或者删除操作时有信息提示该操作的后果；

（4）设置少量快捷键，将常用的功能更方便用户进行调用；

图12是记分员查询运动员比赛记录的窗口，在记分员输入一个字后会显示提示输入信息，记分员可以点击提示信息快速完成输入。其他类似的查询功能均有该功能，辅助记分员操作该系统。图13是系统默认支持的快捷方式，这些快捷方式都是用户较为频繁调用的功能。有了这些快捷方式，用户不需要进到具体的菜单栏便可快速启动该项功能。

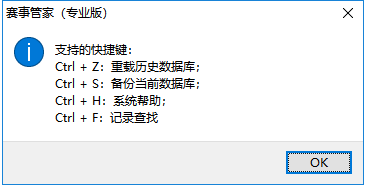
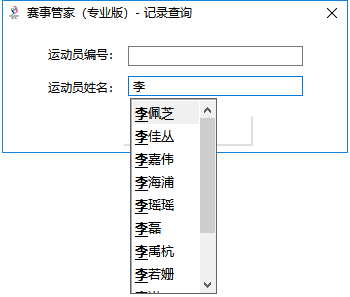


图 12搜索自动补全功能 图 13客户端版支持的快捷键

### **3.3.4 系统帮助**

（1）鼠标在按钮处停顿会显示提示文字，告知用户该按钮的功能；

（2）内置智能帮助工具，用户输入功能模块就可了解到该功能如何操作；

（3）丰富的用户手册供用户下载阅读；

图14是系统内置的“帮助系统”，便于软件使用者在碰到不会使用的功能时快速地查找相关的功能文档。图15是按钮的气泡提示功能，当用户不清楚某个按钮的具体操作时，可以将鼠标在该按钮上停留2秒，系统会显示关于该按钮的帮助信息。

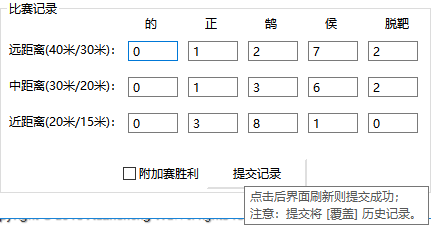
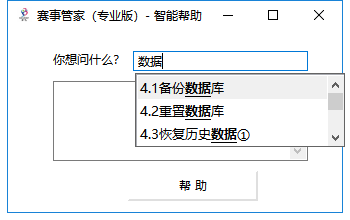


图 14带提示信息的智能帮助 图 15按钮功能的气泡提示

### **3.3.5 系统响应**

考虑到请求响应时长对用户的反馈，采取文字提醒、进度条提醒等方式进行反馈。 （1）0-2秒的相应操作，系统不会做出提示；

（2）2-5秒的操作系统会在任务开启前进行“长耗时警告”；

1. 对于5秒以上的任务使用多线程的方式实现并行运行，同时显示进度条；

## 3.4 网页端界面设计

DAESS的网页端主要满足教练员与运动员的三个需求：获取赛事信息、自主在线报名和在线成绩查询。为了将“信息”本身作为核心凸显出来，提高运动员群体搜集信息的效率并更好地支持移动端设备，网页端界面设计秉持扁平化设计理念。

### **3.4.1简洁的设计元素**

1. 优先呈现重要赛事信息；
2. 放弃一切装饰效果，所有元素边界干净利落；

图17为本系统网页端首页，比赛时间、地点、报名和成绩查询链接等重要信息清晰地罗列在页面中。此外，参赛队员可以点击“下载”按钮，一键获取更多详细的赛事文件。



图 17 DAESS网页版首页

### **3.4.2 简单的交互设计**

1. 重要信息漏填提示；
2. 在共性中寻找特性，相关内容规划在同一页面；
3. 轻松切换成绩展示组别标签；

图18为报名界面，在教练员信息漏填的情况下，网页会自动提示用户补充填写该字段，避免了信息收集不全的问题。同时，所有的填报相关信息都会在这个页面的后半部份要求填写，免去了用户多个页面来回切换的困难。



图 18在线报名页面

图19为成绩公布页面，页面中各轮次比赛的标签可帮助用户在不关闭网页的情况下，查看不同组别轮次比赛的实时成绩。另外，点击表格中的字段，页面会按照该字段进行排序。



图 19 在线成绩查询页面

# 测试报告

## 4.1测试说明

测试目的在于发现本系统潜在的功能性问题，并且评估靶位编排、成绩录入等长耗时任务效率是否满足设计需求。本测试分为单元测试、内部测试和公开测试三个阶段，覆盖界面设计、实用性、功能性、稳定性、性能表现和数据安全六大方面。

## 4.2 测试环境及时间安排

测试流程共计13周，共发布了16个测试版本，其中内测12个版本，公测4个版本。内测阶段的测试环境为表3所示配置；测试任务的安排情况见表4。

表 3 测试环境表

|  |  |  |
| --- | --- | --- |
| **用途** | **硬件环境** | **软件环境** |
| Web服务器 | CPU: Intel E5-2682 | 操作系统: Windows Server 2012 |
| 内存：4G | 应用程序：Python、Flask、Sqlite3 |
| 硬盘：40G | 数据库：SQLite3 |
| DB 服务器 | CPU:Intel E5-2682 | 操作系统:Windows Server 2012 |
| 内存：4G | 应用程序：Python、mysqldb |
| 硬盘：40G | 数据库：MySQL |
| 客户端 | CPU：Intel i7-6560U | 操作系统：Windows 10 |
| 内存：8G | 应用程序：DAESS、Excel |
| 硬盘：256GB | 数据库：SQLite3 |

|  |  |  |  |
| --- | --- | --- | --- |
| **任务名称** | **用时** | **开始时间** | **说明** |
| 单元测试 | 6 周 | 2018-6-15 | 使用少量模拟数据逐个功能测试 |
| 内部测试 | 2 周 | 2018-8-1 | 团队成员使用小型比赛历史数据测试 |
| 压力测试（公测） | 5 周 | 2018-8-15 | 招募志愿者使用大型比赛历史数据测试 |

表 4 项目测试进度安排表

## 4.3 主要修复的测试项目

表 5 测试中发现的部分严重系统漏洞

|  |  |  |
| --- | --- | --- |
| **故障描述** | **错误诊断** | **处理方法** |
| 按照运动员姓名查询信息失败 | 部分姓名中的生僻字在G2312编码字库里不存在；出现男女同名情况； | 1.修正为UTF-8编码格式；  2.增加性别筛选条件； |
| 比赛结果同分运动员无法按照附加赛成绩排名 | 数据库的成绩记录表中没有附加赛字段 | 1.修改表结构，增加附加赛字段； |
| 导出成绩册中部分运动员中文名处显示为乱码 | 不同电脑的默认编码格式不同，导致数据读取的时候崩溃 | 1.强制输出文件的格式为UTF-8； |
| 编排靶位任务超时 | 算法复杂度过高；内存优化较差导致低性能设备负担过重 | 1. 重写编排靶位的算法； 2. 使用DaPy作为数据操作框架，其对于部分操作进行了内存优化； |
| 无法启动系统 | 旧设备的CPU性能较差，系统的动态背景导致70%的CPU占用率；旧设备的内存容量较小，系统的缓存数据较多 | 1. 调整动态背景逻辑，使其基于硬件配置自适应调整更新频率； |

## 4.4 测试结果

最后一次压力测试的结果表明，系统所有性能、功能均符合设计目标，满足赛事主办方对于关于性能和功能的需求。下表6是DAESS最后的性能测试表现。

表 6 性能测试结果

|  |  |  |  |
| --- | --- | --- | --- |
| **测试实例** | **测试过程描述** | **测试结果** | **设计目标** |
| 编排功能 | 将历史比赛成绩录入系统进行靶位编排，比较与历史编排的结果是否有差异并计时 | 0.2秒 | 1秒 |
| 数据传输 | 基于“联机工作”功能在手机热点搭建的局域网中同步赛事数据 | 20MB/秒 | 1MB/秒 |
| 数据同步 | 将大型比赛（300条）历史数据导入数据库 | 3秒 | 5秒 |
| 风格一致 | 15名志愿者体验产品后给出风格是否统一的评价，统计认为“风格一致”的人数 | 12人 | 9人 |
| 成绩录入 | 志愿者限时录入纸质成绩，统计录入的记录数 | 10条 | 8条 |
| 适配低性能设备 | 选用i3处理器的笔记本电脑，检测DAESS运行期间的CPU平均占用率 | 5% | 30% |
| 成绩查询并发能力 | 编写爬虫程序，对开放的成绩查询网站地址无限次HTTP长链接请求，统计系统崩溃峰值的TCP链接数 | 2000链接 | 500链接 |
| 数据备份 | 在系统运行期间强制重启电脑，重启后检查是否无法启动或启动后数据丢失 | 备份正常 | 备份正常 |
| 适配弱网络环境 | 通过“导出数据”功能将数据导出，U盘拷贝后使用“导入数据”功能合并数据是否成功 | 同步正常 | 同步正常 |
| 数字签名 | 修改待导入数据中运动员的成绩，验证系统是否能识别这一非法改动 | 识别 | 识别 |

## 4.5 本项目应用成果

上述测试中，本系统表现出的优势使其快速被上海市大学生体育协会、上海对外经贸大学体育部等单位认可，短时间内顺利支持了多场大型射箭锦标赛。

首先，本系统在2018年5月就以赛事编排、成绩录入、成绩报表三个功能支持了 80 余名运动员的“2018 中国大学生射箭（射艺）锦标赛——预选赛”。此次预选赛同时还有另一套基于B/S架构的比赛管理系统——拓世助手。即使在比赛中由于网络问题导致“拓世助手”系统瘫痪，DAESS却依然稳定地服务着整场比赛。本系统的稳定性成为组委会最终选择本系统支持2018年11月的锦标赛的重要原因。

2018年11月，其顺利支持了由中国大学生体育协会主办，超过 400 名运动员参加的“2018 第二届中国大学生射箭（射艺）锦标赛”。图16是DAESS对此次锦标赛支持过程的时间轴。DAESS在本次比赛中，“线上报名”功能共采集 72 只代表队超 410 名运动员的报名信息；“靶位编排”安排了93场比赛的的运动员靶位信息，“在线成绩查询”功能同时在线人数的峰值达到 1430 人，自动编制的成绩册共计33页；赛事组委会对本系统给予了高度评价。

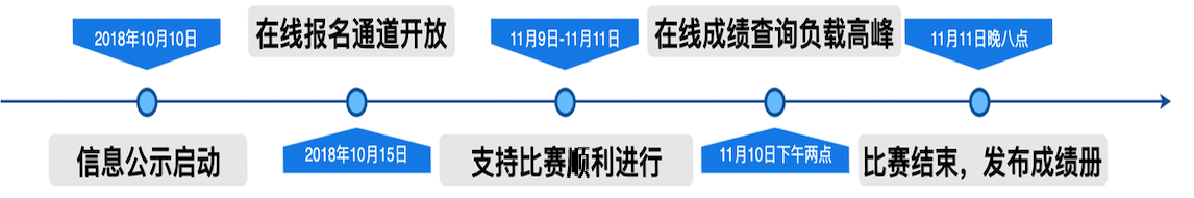


图 16 DAESS系统支持2018中国大学生射箭射艺锦标赛

# 安装及使用

**5.1 安装过程**

（1）请确保您的计算机使用 Windows 64 位操作系统;

（2）前往官方网站http://daess.kitgram.cn下载最新版的DAESS;

（3）运行下载好的exe文件，如果您下载的是Pro客户端，系统提示输入授权码;

（4）将软件上的申请码发送邮件至kitgram@163.com，我们会尽快与你取得联系;

（5）输入授权码后，请您注册管理员用户（用户名/密码）;

（6）安装完成，进入系统，祝您使用愉快!

**5.2 客户端典型使用流程**

（1）同步数据库：在首页点击“终端管理”->“导入数据”->选择安装路径下“实例数据”文件夹当中的athlete\_Info.bat文件 -> “导入”。

（2）录入排位赛成绩：在首页点击“成绩录入”->“排位赛”->在成绩录入界面左侧运动员编号中输入“1a”(代表排位赛中1号靶位第1顺位的运动员）->“敲下回车”->系统左侧会补全该运动员的信息，在右侧随意的输入一些数据后->“敲下回车”。至此，您已经成功输入了一名运动员的比赛记录。

（3）导出排位赛成绩：在首页点击“记录管理”->“记录导出”->“排位赛成绩册”，然后进入安装路径下的“Files/Scores”文件夹下即可找到“个人排名赛总成绩表.xls”。

**5.3 网页端典型使用流程**

（1）在线报名：登入赛事官方网站(http://www.kitgram.cn/ucac)->“在线报名”->在此页面中输入报名的运动员、领队和教练员的信息，在页面底部点击“现在报名”->进入领队的邮箱，查收系统发送的报名确认邮件，核对邮件中附上的报名信息是否有误，如果无误则点击“确认报名”。

（2）成绩查询：登入赛事官方网站(http://www.kitgram.cn/ucac)->“成绩查询”->在呈现的新界面中可以选择相应的赛程，并可以选择不同的字段进行排序。

# 项目总结

分布式射箭比赛管理系统（DAESS）从无到有历时一年时间，其中的过程不可谓不艰辛但也让团队收获颇丰。我校作为一所商科院校在计算机方面的各项资源都较为有限,所以从立项开始，所有技术难题都是通过团队成员自学解决的。从客户端的UI设计及wxPython实现、网页前端的HTML和CSS制作到服务器后端的Flask编写与数据库部署，团队历经了数不清难熬的夜晚、阅读了大量的资料。通过这个项目我们收获的不仅仅是那几千行代码，更是代码背后计算机科学的思维能力和实际的问题解决能力。

在产品的开发过程中，我们是自己的产品经理，因此团队有机会把课内的商业知识运用于实践中。团队通过做竞品分析，发现了市面上主流的产品都有各自的优缺点。与此同时，团队通过问卷调查的方式向射箭运动员、教练员收集意见。我们要求自己取长补短、把握痛点，做出一款真正能解放劳动力的产品。

DAESS解决方案的设计思想被我们称为“B/S + C/S 混合部署架构”。从最基本的单节点模式开始，一台搭Pro的个人电脑，便可以支持一个中小型规模比赛。为了提高效率，通过增加Lite版客户端构造本地C/S架构群。在交互渠道方面，系统通过额外部署的DB服务器和Web服务器形成的B/S架构群搭建了组委会与运动员之间沟通的新桥梁。在广域网连接状况较好时，C/S架构群中的Pro端与B/S架构群中的 DB 服务器进行数据同步，让运动员可以实时查询到最新的比赛数据。

在未来，团队打算应用新技术进一步提高用户体验。一方面，在成绩录入功能中应用OCR技术，通过图片扫描纸质成绩中的手写数据。其次，团队拟搭建运动员与教练员的知识图谱，使运动员可以轻松地了解到队友、对手的情况。另外，随着更多功能的加入导致当前客户端过于沉重，还可将现有的客户端重构为Chromium Embedded Framework配合Micro Service的架构。

1. DaPy是一个Python数据处理/分析框架，由上海对外经贸大学统计与信息学院的学生开发，[项目网站](http://dapy.kitgram.cn/)。 [↑](#footnote-ref-0)