## 广东轻工职业技术学院-信息技术学院 评分互动平台

**项目需求说明书**

广州小马信息科技有限公司  
2017-08-07

E:\01-标准资料\03-品牌标准文档\03-盛世VI手册\金碟色块-红.tifE:\01-标准资料\03-品牌标准文档\03-盛世VI手册\金碟色块-绿.tifE:\01-标准资料\03-品牌标准文档\03-盛世VI手册\金碟色块-蓝.tif

社交网络、移动互联、云计算三大技术将深刻改变管理世界

文档控制

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [√] 正式发布  [ ]正在修改 | 文件标识： | 广轻小马实训课程 |
| 历史版本： |  |
| 当前版本： | V1.0 |
| 作 者： | 黄桐鑫 |
| 完成日期： | 2017-08-07 |

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [√] 正式发布  [ ]正在修改 | 文件标识： | 广轻小马实训课程 |
| 历史版本： |  |
| 当前版本： | V1.0 |
| 作 者： | 黄桐鑫 |
| 完成日期： | 2017-08-07 |

## 文档目录

[广东轻工职业技术学院-信息技术学院 评分互动平台 1](#_Toc491935929)

[文档目录 4](#_Toc491935930)

[1 引言 5](#_Toc491935931)

[1.1 编写目的 5](#_Toc491935932)

[1.2 文档约定 5](#_Toc491935933)

[1.3 读者对象 5](#_Toc491935934)

[1.4 项目背景 5](#_Toc491935935)

[1.5 项目范围 5](#_Toc491935936)

[2 系统概述 6](#_Toc491935937)

[2.1 整体流程概述 6](#_Toc491935938)

[2.2功能性需求 6](#_Toc491935939)

[3 硬件需求 6](#_Toc491935940)

[3.1 服务器配置 6](#_Toc491935941)

[4. 项目简介 6](#_Toc491935942)

[5. 评分互动平台业务 6](#_Toc491935943)

[6 详细功能需求 7](#_Toc491935944)

[6.1 用户管理 7](#_Toc491935945)

[6.2 展品管理 7](#_Toc491935946)

[6.3 评论管理 8](#_Toc491935947)

[6.4 评分管理 8](#_Toc491935948)

[6.5 提问管理 8](#_Toc491935949)

[6.6 送礼管理 8](#_Toc491935950)

[6.7 抽奖管理 9](#_Toc491935951)

[7 建设原则和技术线路 9](#_Toc491935952)

[7.1 性能需求 9](#_Toc491935953)

[7.2 技术线路 9](#_Toc491935954)

## 1 引言

### 1.1 编写目的

本文档的编写目的是评分互动平台的开发提供：

* 软件总体要求，作为用户和软件开发人员之间了解的基础；
* 功能、性能和接口的要求，作为软件人员进行设计和编码的基础
* 验收标准，作为用户确认测试的依据。

### 1.2 文档约定

本需求规格说明书的主要内容有：介绍产品的背景，概述产品的功能需求、环境需求、性能需求、平台角色等内容。

### 1.3 读者对象

该需求规格说明书提供给最终用户、研发人员、测试人员等相关人员阅读。

### 1.4 项目背景

随着科学的进步，展品的展示及评分方式多种多样化，但如今的人们的生活大部分离不开互联网。为了方便评分者之间的线上交流及评分，实现评分者及有兴趣的人更多的参与进来，需要一个全民互动的线上平台来支撑。特编写此文档以梳理需求，明确内容。

### 1.5 项目范围

“评分互动平台”项目用于企业、学校或者其他机构对参展项目进行评分及互动。

## 2 系统概述

### 2.1 整体流程概述

### 2.2功能性需求

## 3 硬件需求

### 3.1 服务器配置

|  |  |  |
| --- | --- | --- |
| **配置项** | **最低配置** | **标准配置** |
| 操作系统 | Centos 6.8及以上 64位 | Centos 6.8及以上64位 |
| 处理器 | 2核 | 4核 |
| 内存 | 4GB | 8GB |
| 硬盘 | 40GB | 60GB |

## 4. 项目简介

1.后台对前端需要的信息进行管理。

2.前端为手机端，有用户注册登录、个人资料、展品列表，评分者可选择自己有意愿的展品进入交流平台。主要操作有文字交流、送礼物、评分及抽奖功能。

## 5. 评分互动平台业务

1、用户使用手机号码注册账号，登录。有个人资料管理，参展列表等页面。

2、参展者在前端提交参展项目的几张图片录入系统及描述等提交审核，由后台管理者审批。

3、前端的展品展示及轮播。

4、用户可以看到参展列表，点击进入参展详情时需要进行登录。登录之后可参与完整的功能。

5、互动功能：

评论互动区：参与者可发表评论，评论以弹幕及普通内容展示及赠送礼物功能并获得热度。

评分区：由后台录入的规则进行评分，登录用户可以参与评分功能，该功能主要通过一些单选或者多选题来进行评分。

答题区：指定等级的用户可以提出问题，由其他参与者参与答题。

6、参展详情页面显示参展者信息，及该项目的热度。热度由观看者、礼物、及评分题来统计。最终展品的分数由热度来决定。

7、抽奖功能：参展结束之前，用户可在抽奖页面点击参与抽奖，参展结束的时候由特定等级的用户操作开奖。

## 6 详细功能需求

### 6.1 用户管理

1：前端

用户使用手机号进行注册，注册后的用户可使用验证码或者密码进行登录。

页面及操作：

注册页面

登录页面

重置密码

个人中心

密码修改

手机号绑定修改

修改资料（头像，学历，地区等）

我的参展（筛选：待审核，带参展，参展中，参展结束）

展品详情（可以查看相应的评论、参与者的答题情况内容、提问情况、送礼情况、奖项设置、热度等）

新增参展申请（参展内容，图片，奖项设置）

2.后台：

用户管理，可对用户进行拉黑，取消拉黑操作

页面及操作：

用户管理页面

### 6.2 展品管理

1.前端

用户登录之后可以申请展品参展，审核通过的展品，参展者可以在开始参展的时候上线参与参展。上线参展结束之后的展品不得再次参展。

页面及操作：

待参展（展品封面，简介等）

参展中（展品封面，简介等）

点击进入参展详情页（展品轮播图片及热度等信息，互动区）

参展结束（列表显示参展结束的展品，热度等详细信息）

2.后台

后台对前端用户申请的展品进行审核。当参展项目已经上线参展，要对用户与展品的互动数据进行操作。例如参与者、礼物、评分等对热度的影响。

可以查看相应的评论、参与者的答题情况内容、提问情况、送礼情况！

页面及操作：

待审核展品列表（详情，录入评分规则（只有录入评分规则才能通过），驳回（驳回意见））

待参展展品列表（详情）

已参展展品列表（详情-包括热度等）

### 6.3 评论管理

1.前端

参展详情页面有评论功能。评论以弹幕跟普通文本形式展现

2.后台

在展品管理可以查看相应展品的评论列表

### 6.4 评分管理

1.前端

参展详情页面有评分功能。参与者可以参与评分（选择题，多选题等形式）

2.后台

审核参展时录入评分规则

### 6.5 提问管理

1.前端

特定等级的用户有权利在参展途中提问，在场人员可以参与解答

2.后台

在展品管理可以查看提问列表及解答情况

### 6.6 送礼管理

1.前端

在评论互动页面用户可以赠送礼物，礼物以什么形式展现出来由设计开发人员决定。

2.后台

后台对可赠送礼物种类及赠送了礼物增加的热度等进行管理。

### 6.7 抽奖管理

1.前端

参与抽奖及开奖页面

## 7 建设原则和技术线路

### 7.1 性能需求

系统设计遵循以下原则：

* + 先进性

系统采用多层结构设计，采取XML技术进行数据的存储、传输。

* + 经济性

避免重复建设，力争节约资金，对现有的办公网络充分利用、发挥效益。

* + 扩展性

系统充分考虑今后功能扩展需要，预留了各种扩展接口。

* + 简单性

系统功能完善，但操作维护便利，界面友好，尤其对单位系统管理员而言，无须复杂的技术培训和繁琐的编程，即可对应用流程进行调整和维护。

* + 互联性

该系统可以与其它信息系统实现无缝联接，可以与各种关系型数据库平滑联接。

* + 可靠性

由于系统的特定用户群，在系统平台上将保证系统的高保密性、高安全性和高可靠性。系统设计中，采用可靠成熟的技术和平台产品，数据有适量冗余及其他保护措施，平台和应用软件具有容错性、健壮性等。

* + 安全性

系统采用多重安全措施确保系统多层次的安全需求。

### 7.2 技术线路

MVC开发模式

严格采用“模型-视图-控制器”（Model-View-Controller）开发模式，把业务逻辑封装在Model层中，显示逻辑放在View层中,而在View层中所能访问Model的信息完全由控制层（Controller）控制。Model提供统一的对外接口，而不管View端是基于什么样的开发语言。

MVC是一种多种设计模式组合而成的设计模式，指的是一种划分系统功能的方法，它是Model-View-Controller的缩写，最早是由Xerox（施乐）在20世纪80年代为Smaltalk-80语言发展提出的。随后，它成为了一种著名的用户界面设计架构。

在MVC经典架构中，它把应用程序（也可看做是应用程序的一个模块）分为3个部分：

**1．模型（Model）**

模型代表了该应用程序的核心功能，它负责处理用户的数据，实现业务逻辑。同时，模型还为视图的显示提供数据，并可被多个视图所共用。

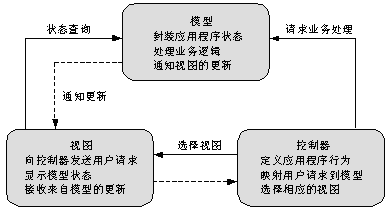
**2．视图（View）**

视图主要指与用户交互的界面，也即应用程序的外观。视图可以接收用户的输入，但它并不包括任何实际的业务处理，它只是将数据转交给控制器。同时，视图还负责展现模型传递给用户的数据，当后台模型更新数据时，视图也应当随之更新它的显示。

**3．控制器（Control）**

控制器负责接收用户的请求和数据，接着它会做出判断将请求和数据交由哪一个模型来处理，最后调用视图来显示模型返回的数据。

它们之间的关系如下图所示。

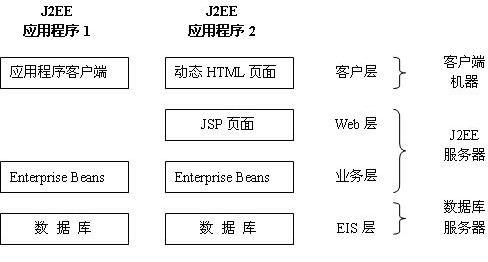


基于J2EE架构

充分利用JAVA语言的跨平台性，同时遵照SUN公司的J2EE架构标准。

J2EE是一个开放的、基于标准的平台，可以开发、部署和管理N层结构的、面向Web的、以服务器为中心的企业级应用，它是利用Java 2 平台来简化与多级企业解决方案的开发、部署和管理相关的诸多复杂问题的应用体系结构。

J2EE平台采用一个多层次分布式的应用模式。这意味着应用逻辑根据功能被划分成组件，组成J2EE应用的不同应用组件安装在不同的服务器上，这种划分是根据应用组件属于多层次J2EE环境中的哪一个层次来决定的。J2EE应用可以由三或四个层次组成，J2EE多层次应用一般被认为是三层应用，因为它们是被分布在三个不同的地点：客户端机器、J2EE服务器和数据库或后端的传统系统服务器。三层架构应用是对标准的客户端/服务器应用架构的一种扩展，即在客户端应用和后台存储之间增加一个多线程应用服务器。



J2EE多层应用

J2EE体系包括JSP、Servlet、EJB、WEB SERVICE等多项技术。这些技术的出现给电子商务时代的WEB应用开发提供了一个非常有竞争力的选择。怎样把这些技术组合起来，形成一个适应项目需要的稳定架构是项目开发过程中一个非常重要的步骤。

一个成功的软件需要有一个成功的架构，但软件架构的建立是一个复杂而又持续改进的过程，软件开发者们不可能对每个不同的项目做不同的架构，而总是尽量重用以前的架构，或开发出尽量通用的架构方案，Struts就是流行的基于J2EE的架构方案之一，其他常用的基于J2EE的架构方案还有Turbine、RealMothods等。