易拉罐学习论坛设计规格说明

Ver:1.0

**目 录**

[1 引言 3](#_Toc301301027)

[1.1 目标 3](#_Toc301301028)

[1.2 文档范围 3](#_Toc301301029)

[1.3 术语和缩略语 3](#_Toc301301030)

[1.4 参考资料 3](#_Toc301301031)

[1.5 系统目标和约束 3](#_Toc301301032)

[2 系统设计 4](#_Toc301301033)

[2.1 系统架构概述 4](#_Toc301301034)

[2.2 对象模型 4](#_Toc301301035)

[2.3 接口 4](#_Toc301301036)

[2.4 特性实现 4](#_Toc301301037)

[2.5 错误代码 9](#_Toc301301038)

[2.6 错误日志 9](#_Toc301301039)

[2.7 部署视图 9](#_Toc301301040)

[3 数据库设计 11](#_Toc301301041)

[3.1 逻辑模型 11](#_Toc301301042)

[3.2 物理模型 11](#_Toc301301043)

[4 质量及其他方面 12](#_Toc301301044)

[4.1 可维护性 12](#_Toc301301045)

[4.2 安全性 12](#_Toc301301046)

[4.3 可扩展性 12](#_Toc301301047)

[4.4 可靠性 12](#_Toc301301048)

[4.5 可用性 12](#_Toc301301049)

[4.6 性能设计 13](#_Toc301301050)

[5 附录 13](#_Toc301301051)

[5.1 附件 13](#_Toc301301052)

[5.2 修过记录 13](#_Toc301301053)

# 引言

## 目标

在完成软件开发前期的准备工作如项目需求等，结合《需求确认书》，，项目小组提出了这份软件设计说明书。

此概要设计说明书对易拉罐学习论坛系统的功能分配，模块划分，程序的总体结构，输入输出和接口设计，运行设计，数据结构设计及出错设计等方面作了全面的概括性的说明，为软件详细设计奠定了基础，同时作为系统分析员工作的阶段性总结和程序员进行开发及未来测试的重要文档资料。

## 文档范围

本文档包括系统的架构设计、数据库设计、安全性、可靠性、可用性等方面的规划和设计。

## 术语和缩略语

## 参考资料

部分内容参考了《易拉罐学习论坛需求确认书》

## 系统目标和约束

系统目标：项目需要用户登录、注册、个人信息、以及对帖子的管理功能。

系统的约束：因为时间有限，该系统只实现最基本的功能。

# 系统设计

## 系统架构概述

该系统架构采用Laravel框架，操作、数据、视图分层

## 对象模型

该系统采用面向对象的设计方法。

## 接口

*[详细说明本系统内部每个接口的每个方法的定义。*

*注意：如果本系统需要和其他系统交互，则应该将交互的接口协议单独成册，而不是写入本节中。]*

#### Home模块

首页：

Class IndexController extends Controller(){

Public function index(){} //显示首页信息

Protect function latestDynamic(){} //获取最新动态资源

Protect function specialRecommendation(){} //获取特色推荐资源

}

用户管理模块：

class UserManagerController extends Controller{

Public function loginPage(){} //拉取用户登录页面

Public function login(){} //登录

Public function registerPage(){} //拉取用户注册页面

Public function register(){} //注册

Public function exit(){} //退出

Public function rememberPassword(){} //记住密码

Public function forgetPassword(){} //忘记密码

Public function Info(){} //显示个人信息

Public function modifyInfoPage(){} //拉取修改个人信息页面

Public function modifyInfo(){} //修改个人信息

Public function modifyPasswordPage(){} //拉取修改密码页面

Public function modifyPassword(){} //修改密码

Public function GZHList(){} //显示用户发帖信息列表

}

管理模块：

Class ProjectManagerController extends Controller{

Public function centerPage(){} //拉取发帖中心主页面

Public function create(){} //新建帖子

Public function deletePost(){} //删除帖子

Public function listPost() //返回帖子列表信息

Public function info(){} //显示帖子详情

Public function comment(){} //评论帖子

Public function listComment(){} //返回帖子评论列表信息

Public function deleteComment(){} //删除帖子评论

}

#### Admin模块

轮播图管理：

class ShufflingImgController extends Controller{

Public function add(){} //新增

Public function delete(){} //删除

Public function list(){} //显示图片列表

}

推荐帖子：

Class LatestDynamicController extends Controller{

Public function add(){} //新增

Public function delete(){} //删除

Public function list(){} //显示帖子列表

}

分类推荐：

Clsss SpecialRecommendationController extends Controller{

Public function add(){} //新增

Public function delete(){} //删除

Public function list(){} //显示分类列表

}

帖子列表页：

Class ProjectDisplayController extends Controller(){

Public function add(){} //新增

Public function delete(){} //删除

Public function list(){} //显示展示的帖子列表  
}

用户管理

Class userManagerController extends Controller(){

Public function list(){} //显示用户及其基本信息列表

Public function listPost(){} //显示用户所有帖子信息列表

Public function userCount(){} //返回用户个数

Public function postCount(){} //返回公众号个数

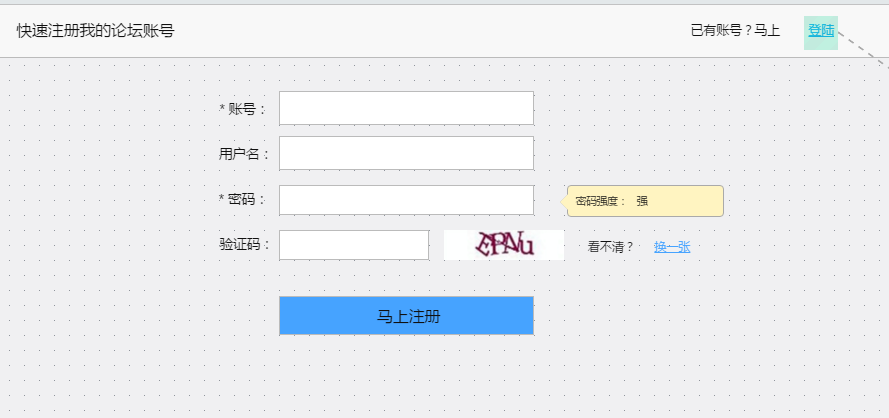
Public function userInfo() //显示用户详情

Public function postInfo(){} //显示公众号详情

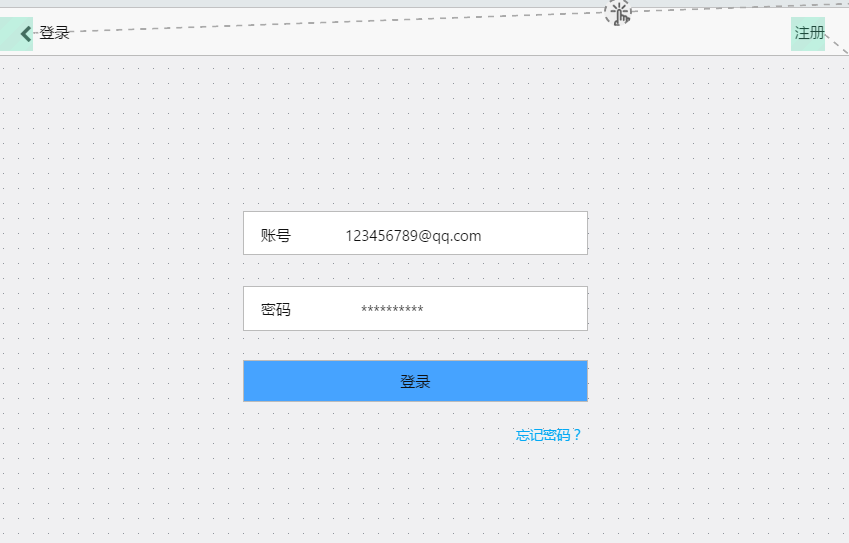
}

## 特性实现

1. 用户注册
   1. 用户注册页面
   2. 注册成功，跳转至首页



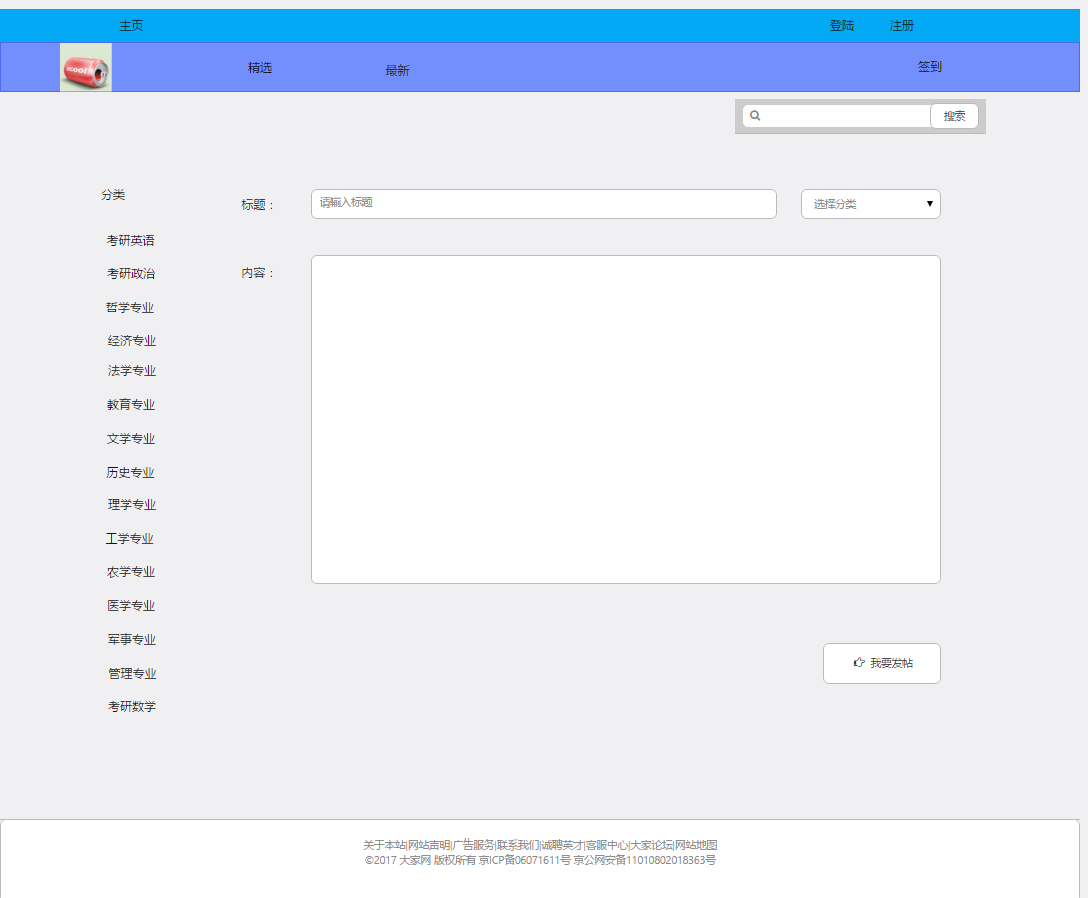
1. 用户登录



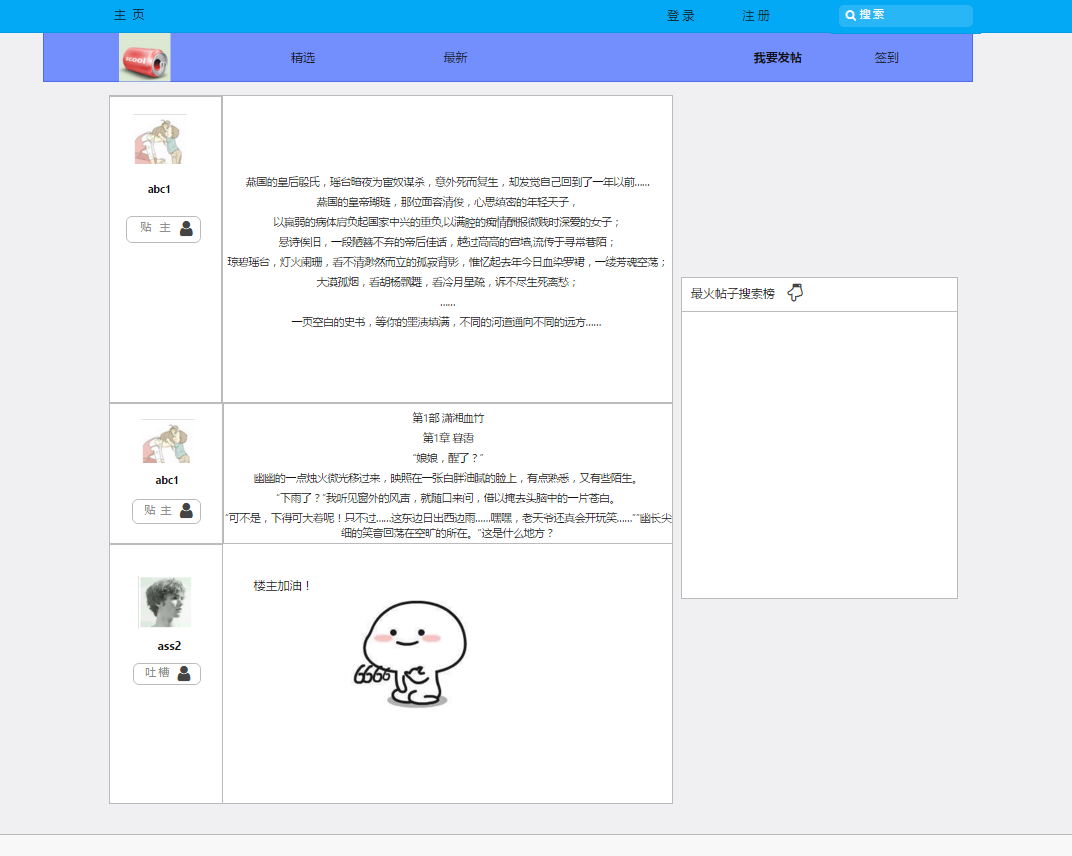
3.用户选择发帖

* 1. 展示帖子列表
  2. 

用户发表帖子



用户查看帖子详情



## 错误代码

1. 项目新建信息错误处理
2. 开发接口错误信心转义输出

## 错误日志

参数错误，服务器错误，请求错误等

## 部署视图

1. 硬件环境：
   1. 需要互联网
   2. 至少需要一台服务器
2. 软件：
   1. 采用centOS 6.4
   2. 需要Apache2.2作为WEB服务器。
   3. 需要安装PHP 7.0.12解释器
   4. 需要安装mysql服务器
3. 配置：
   1. 因为PHP5.3以上的版本已经不支持mysql扩展，所以需要自行下载微软提供的The SQL Server Driver for PHP(SQLSRV20.EXE)
   2. 下载后解压缩，将所有的.so文件拷贝到php安装目录的ext文件夹中，然后打开php.ini，添加以下语句打开php\_sqlsrv和php\_pdo\_sqlsrv扩展：

[Dynamic Extension]

extension=php\_pdo.so

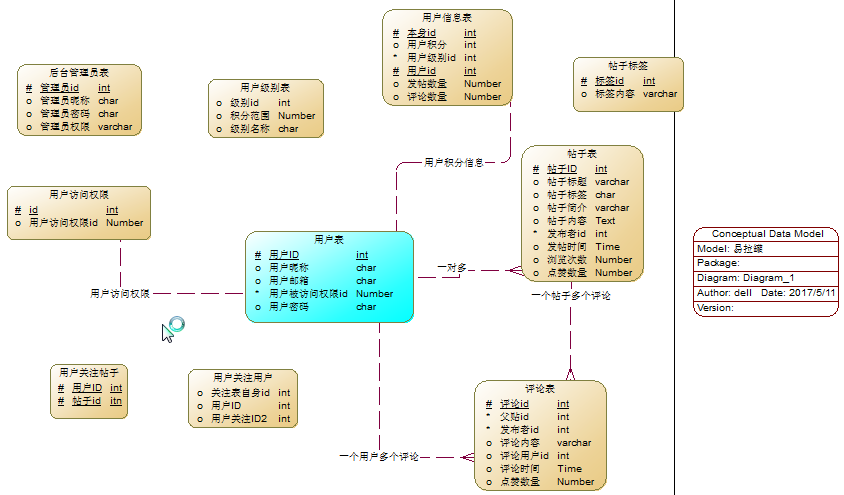
extension=php\_mysqli.so

* 1. 安装mysql服务 yum install -y mysql-server

1. 部署：
   1. 配置PHP.ini,关闭错误提示。
   2. 配置PHP连接MS SQL的文件，修改IP地址、数据库的用户名和密码。
   3. 将程序包放在Apache安装目录中得/var/www/html目录下。

# 数据库设计

## 逻辑模型



## 物理模型

1. 为每个表初始设置300M的存储空间，以10%的大小扩展。
2. 为上传的图片预留空间。

# 质量及其他方面

## 可维护性

1. 通过程序注释等方式增加代码的可读性和可维护性。
2. 将数据访问层分离，做成一个个函数，由其它层调用，以增加代码的可维护性。
3. 数据库有日志记录，系统一旦出现故障有恢复到故障之前的信息和数据的能力

## 安全性

1. 密码使用md5加密
2. 对用户的输入进行验证
3. 对用户的输入的特殊字符进行转义，更采用laveral框架防止sql注入攻击

## 可扩展性

1. 可以通过增加硬件资源的方式提高系统的响应速度。
2. 可以通过修改代码，扩充系统的功能。

## 可靠性

1. 数据库按时备份，有日志记录

## 可用性

1、做好充分的前期工作，系统一旦投入使用，尽量减少宕机的次数和时间

## 性能设计

[系统性能通常使用事务处理能力或资源利用率来度量。确定系统需求的方法：

识别约束：包括预算、时间、基础结构、可选的开发工具和技术；

确定功能特性：要符合使用场景和用例

确定负载：使用本系统的客户数量]

# 附录

## 附件

[列出支撑本文档的附属文档的名字。如EA文件、绘制系统架构(如有)的viso文件，powerdesigner(如有)文件等]

## 修过记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 修改人 | 修改时间 | 修改内容 | 核准 |
| 1.0 | 刘海燕 | 2011-8-16 |  |  |