

校脸微信公众号设计规格说明

Ver:1.0

目 录

1	引言.....	3
1.1	目标.....	3
1.2	文档范围.....	3
1.3	术语和缩略语.....	3
1.4	参考资料.....	3
1.5	系统目标和约束.....	3
2	系统设计.....	4
2.1	系统架构概述.....	4
2.2	对象模型.....	4
2.3	接口.....	4
2.4	特性实现.....	4
2.5	错误代码.....	9
2.6	错误日志.....	9
2.7	部署视图.....	9
3	数据库设计.....	11
3.1	逻辑模型.....	11
3.2	物理模型.....	11
4	质量及其他方面.....	12
4.1	可维护性.....	12
4.2	安全性.....	12
4.3	可扩展性.....	12
4.4	可靠性.....	12
4.5	可用性.....	12
4.6	性能设计.....	13
5	附录.....	13
5.1	附件.....	13
5.2	修过记录.....	13

1 引言

1.1 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校脸微信公众号的功能分配，模块划分，程序的总体结构，注册，5个模块设计和接口设计，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。

1.2 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。

1.3 术语和缩略语

1.4 参考资料

部分内容参考了《校脸微信公众号需求确认书》

1.5 系统目标和约束

系统目标：项目需要完成头条内容热点推送，大学内标志建筑物介绍记忆基本学校内容介绍等基本功能。

系统的约束：因为时间有限，该系统只实现最基本的功能。

2 系统设计

2.1 系统架构概述

该系统的架构采用两层架构的模式：展示层（视图+业务逻辑）和数据访问层。采用面过程的设计方法。

2.2 对象模型

该微信公众号采用非面向对象的设计方法。

2.3 接口

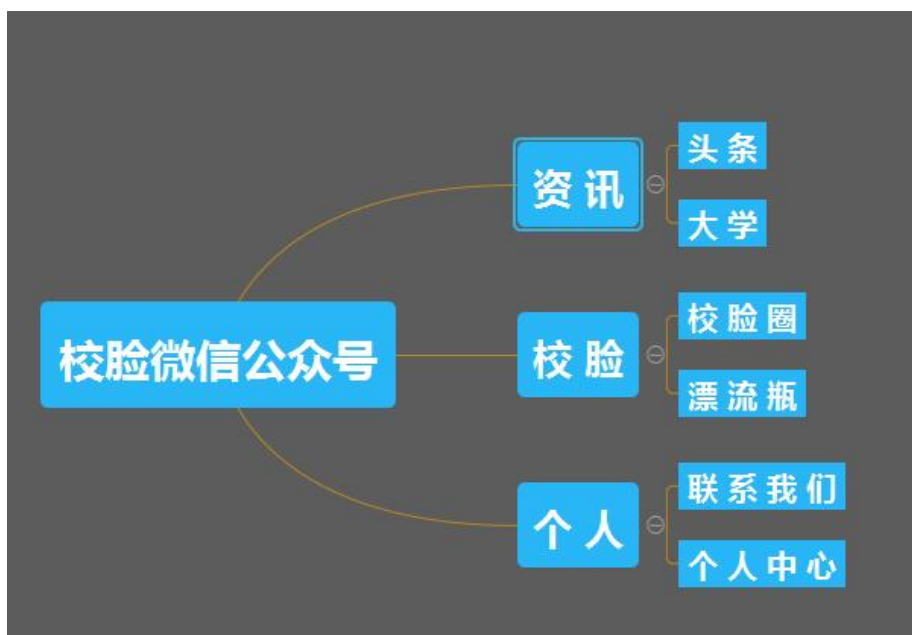
[详细说明本系统内部每个接口的每个方法的定义。

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。]

2.4 特性实现

1、自定义菜单实现

- a) 关注校脸微信公众号，菜单如下



b)

2、资讯----头条推送

a) 后台人员自己从后台添加图文消息

* 标题：	<input type="text"/>
图片：	<input type="button" value="浏览..."/> 1.jpg
内容：	<div style="border: 1px solid #ccc; height: 150px;"></div>
<input type="button" value="提交"/> <input type="button" value="返回"/>	

b) 提交之后的数据保存到 xl_topLine 头条文章表

c) 从 xl_topLine 头条文章表取出相关的图片和文章标题，显示在头条界面

- d) 添加的最新内容会更新到页面初始端
- e) 第一次点击头条是从数据库获取，并且利用 thinkphp 的静态缓存功能将静态页面缓存下来，

静态规则定义

静态规则的定义方式如下：

```
1. 'HTML_CACHE_ON' => true, // 开启静态缓存
2. 'HTML_CACHE_TIME' => 60, // 全局静态缓存有效期（秒）
3. 'HTML_FILE_SUFFIX' => '.shtml', // 设置静态缓存文件后缀
4. 'HTML_CACHE_RULES' => array( // 定义静态缓存规则
5.     // 定义格式1 数组方式
6.     '静态地址' => array('静态规则', '有效期', '附加规则'),
7.     // 定义格式2 字符串方式
8.     '静态地址' => '静态规则',
9. )
```

- f) 每次下拉都会从数据库获取，用 javascript 实现

3、显示头条内容详情页

- a) 从头条显示页跳转到头条内容详情页
- b) 头条显示页是图片和文章标题的超链接，点击超链接，传递文章 id 值，跳转到文章显示界面
- c) 在 thinkphp 中启用路由并定义路由规则，

```
'URL_ROUTE_RULES'=>array(
    'news/:id' => 'News/read',
),
```

4、头条中的加号

- a) 点击加号，弹窗有 3 个按钮，用 javascript 实现
- b) 扔瓶子捞瓶子，管理我的瓶子，用动画来实现

5、大学

- a) 大学的名称及标志，从后台添加

首页 > 高校管理 > 新增高校

*学校名称：	<input type="text"/>
*校徽：	<input type="button" value="浏览..."/> 未选择文件。
学校地点：	<input type="text"/>
学校简介：	<input type="text"/>
<input type="button" value="提交"/> <input type="button" value="返回"/>	

b) 添加完成之后至数据表 xl_university 大学校园表

6、校脸圈动态发表及查看

- 发表校脸圈动态，写完的内容保存到数据表 xl_xiaolianArticle
- 发表之后跳转到校脸圈首页，并显示自己发送的内容
- 校脸圈动态也要采用 thinkphp 框架中的静态缓存
- 下拉刷新，获取最新动态

7、校脸圈首页

- 校脸圈首页有用户名称，具体发表时间，图片和动态前 30 个字，in 1985 see 代表，有 1985 人浏览过



b) In 1985 see 是指有 1985 个人浏览过该动态，采用 ajax 技术实现局部刷新

8、校脸圈评论

A) 多表查询，将评论表和文章表结合起来

B) Thinkphp 中的关联查询，一对多，一篇文章对应多条评论

9、漂流瓶的实现

a)

2.5 错误代码

[详细列出错误处理机制所能处理的所有错误代码及其含义]

2.6 错误日志

[说明能够被处理和记录的错误类型和记录方式。]

2.7 部署视图

1、硬件环境：

- a) 需要互联网
- b) 至少需要一台服务器

2、软件：

- a) 需要 Windows Server 2003 或其他 Server 版的操作系统以安装 SQL SERVER2005 企业版。或者其他操作系统，但是使用 SQL SERVER 就不能使用企业版了，可以换成 EXPRESS 版。
- b) 需要 Apache2.2 作为 WEB 服务器。
- c) 0 需要安装 PHP 解释器

3、配置：

- a) 因为 PHP5.3 以上的版本已经不支持 mssql 扩展，所以需要自行下载微软提供的 The SQL Server Driver for PHP(SQLSRV20.EXE)
- b) 下载后解压缩，将所有的.dll 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：
[PHP_PDO_SQLSRV]
extension=php_pdo_sqlsrv_53_ts_vc6.dll

[PHP_SQLSRV]

extension=php_sqlsrv_53_ts_vc6.dll

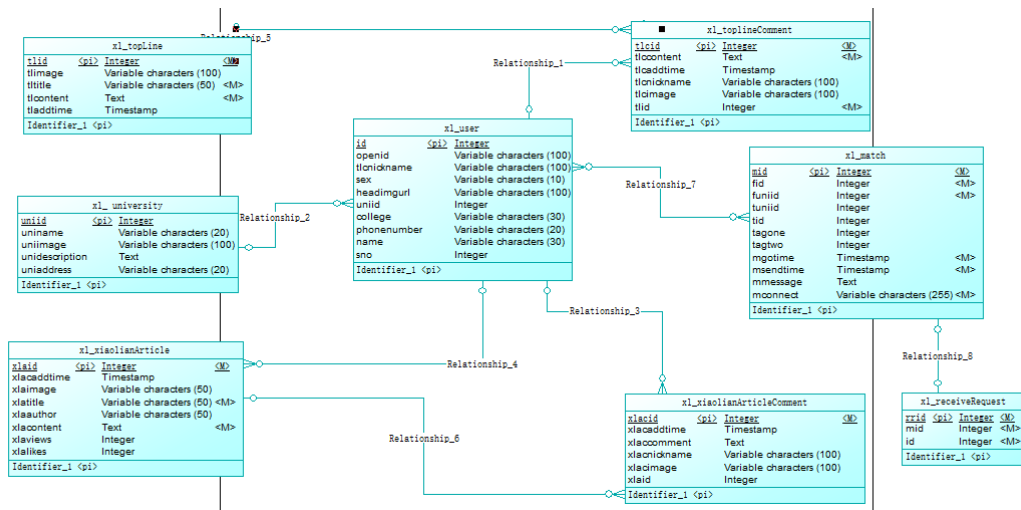
- c) 安装 Microsoft SQL Server 2008 R2 Native Client , 可以去 [http://msdn.microsoft.com/en-us/library/cc296170\(SQL.90\).aspx](http://msdn.microsoft.com/en-us/library/cc296170(SQL.90).aspx) 下载安装, 因为微软的这个扩展包需要它的支持。(注 SQL SERVER 2005 的客户端也不行, 必须要 2008 的)

4、部署：

- a) 配置 PHP.ini, 关闭错误提示。
- b) 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。
- c) 将程序包放在 Apache 安装目录中得 htdocs 目录下。

3 数据库设计

3.1 逻辑模型



3.2 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。

4 质量及其他方面

4.1 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

4.2 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，防止 sql 注入攻击

4.3 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。

4.4 可靠性

- 1、数据库按时备份，有日志记录

4.5 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

4.6 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]

5 附录

5.1 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 visio 文件，powerdesigner(如有)文件等]

5.2 修过记录

版本	修改人	修改时间	修改内容	核准
1.0	郭颖	2016-16-15		