校园+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》，，提出了这份软件设计说明书。

此概要设计说明书对校园+系统的功能分配，模块划分，程序的总体结构，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《校园+需求确认书》

## 系统目标和约束

系统目标：

顾客方面目标：

浏览商品

预定商品

退订商品

预定理发

退订理发

浏览兼职

预定兼职。

商家方面目标：

订单取消

商品上传

商品删除

商品修改

商品查看

理发类型上传

理发时间确定

理发人数限制

兼职机构目标：

上传兼职

系统的约束：因为微信端支付问题未能实现，该系统只实现部分功能，某些功能仍然在探索中。

# 系统设计

## 系统架构概述

该系统的架构采用两层架构的模式：表现层、业务逻辑层、数据访问层和数据存储。采用面向过程的设计方法。

表现层：基于微信的微网站

业务逻辑层：controller控制器

数据访问层：pdo方式，或是直接访问

数据存储：存储在mysql中

## 对象模型

该系统采用非面向对象的设计方法。

## 接口

1、商品管理接口

　　开发者可通过商品管理接口，来增加商品、删除商品、修改商品信息、查询已有商品，并可通过接口对商品进行上下架等操作管理。

2、库存管理接口

　　开发者可通过库存管理接口，来为已有商品增加和减少库存，包括进行与自身系统或其他平台的库存同步。

3、分组管理接口

对已有商品，开发者可通过分组管理接口，来对商品进行分组管理。接口包括增加、删除、修改和查询分组。

4、订单管理接口

　　开发者可按订单状态和时间来获取订单，并对订单进行处理。

5、功能接口

　　目前功能接口暂时只支持上传图片接口一项。微信商户开发接口中所有需要用到图片的地方，都需先使用上传图片接口来预先获得图片的URL。

## 特性实现

1、商家老板登记自己的商品信息

* 1. 登记商品信息页面
  2. 登记成功，可选择修改详细信息
  3. 登记成功，可选择删除详细信息
  4. 登记成功，可选择继续添加商品信息

1. 理发店老板向自己理发店添加发型的相关详细信息
   1. 添加发型的详细信息
   2. 查看自己发型的详细信息
   3. 修改自己发型的详细信息
   4. 删除自己发型的详细信息
   5. 添加做发型的时间段
   6. 添加做发型每个时间段的人数
2. 兼职机构添加兼职相关详细信息
   1. 添加兼职相关详细信息
   2. 修改兼职相关详细信息
   3. 删除兼职相关详细信息
3. 商家老板查看顾客的预定情况
   1. 删除订单
4. 理发店老板查看顾客的预定情况
   1. 查看订单
   2. 删除订单
5. 兼职机构查看顾客的预定情况
   1. 查看顾客预定情况，选择顾客雇佣关系
   2. 删除顾客预定
6. 顾客查看商品
   1. 顾客查看商品信息
   2. 顾客选择预定商品
   3. 删除订单
7. 顾客查看理发类型信息
   1. 查看理发类型信息
   2. 选择订单
   3. 删除订单
8. 顾客查看兼职相关信息
   1. 查看兼职相关信息
   2. 选择兼职类型
   3. 删除兼职订单

## 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]*

## 错误日志

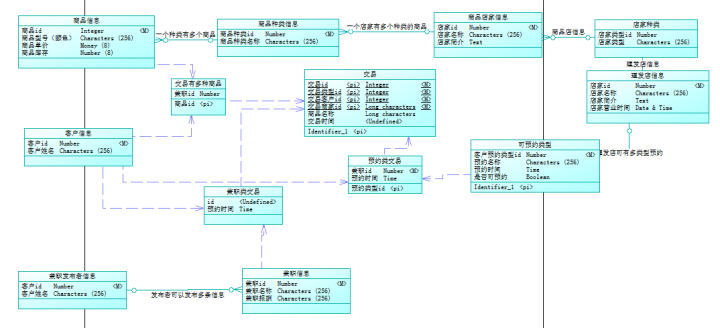
*[说明能够被处理和记录的错误类型和记录方式。]*

## 部署视图

1. 硬件环境：
   1. 需要互联网
   2. 需要一台服务器
2. 软件：
   1. 手机微信，一个微信公众号
   2. 需要apache，mysql，php安装，或者三个的集成环境
3. 配置：
   1. 修改php.ini，使其可以对apache，mysql使用
4. 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www目录中。

# 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，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 | 孙池晔 | 2016-11-20 |  |  |