**天津工业大学**

**毕业设计**

**海豚湾美甲店美业管理系统设计与开发**

**姓 名** 刘 鑫

**学 院** 管 理 学 院

**专 业** 信息管理与信息系统

**学 号** 1110650127

**指导教师** 王 建 正

**职 称** 教 授

2015年6月

1. 绪论

1.1美甲店信息化的发展现状

自二十世纪九十年代以来，通信技术以及计算机技术所带来的技术革命，对信息化和信息技术的发展产生了巨大的推动作用。人们也普遍意识到了信息化的必要性和重要性。这一切都是科技发展的必然结果，不是人们的主观意愿所能改变的，各行各业必将被信息化所渗透，用以提升这些行业的生产力，其急速发展的趋势对所有国家既是严峻的挑战，又是难得的机遇。

就近几年而言，我国的信息化得到了迅猛的发展，信息化水平越来越高，加速了很多传统行业的改造，呈现了传统企业若不实施信息化就有可能被淘汰的局面。并且信息化在国民经济中的地位越来越不可忽视。

我国的企业信息化也越来越普遍，在我们的生活中能随处感受得到。如我们去商场购物，商场会有相应的收银系统对商品的物流库存以及账务流水进行管理；我们去饭店吃饭，饭店会有相应的点餐系统和结算系统对饭店的经营业绩进行管理；我们可以通过互联网在网上购买车票或飞机票，并且能在线选座和支付。由此可见，企业实施信息化会深刻地影响到人们的吃、穿、住、行，为人们的生活带来极大的方便和快捷。企业通过实施信息化不仅可以做到精准、高效的管理，提高生产力，还可以提高企业的经营业绩，降低经营成本，减少不必要的开支，将大部分钱投入到核心业务当中。

就2014年来说，中国线下的美业实体店已经超过300万家，线下的市场规模达到了。1.2万亿。然而整个行业的信息化、互联网化的水平却相当低，另外服务水平也一直被诟病，消费不透明、不诚信等是制约这个行业发展的关键因素。因此，美业行业可以选择实施信息化来帮助实体店升级转型，来解决上述两个痛点。

1.2美业管理系统的开发背景

其他行业一样，能跟得上市场发展趋势的企业，才会面临更多机会。2014年是互联网思维在传统企业的应用元年，业内当今谈论最多的话题也是传统企业如何借助互联网和移动互联网实施信息化。对于美业行业来说，在互联网高度发达的信息化时代，对企业实施信息化已经成为在现代商战中占领制高点的必备武器。然而国内面对美业管理的系统不仅功能参差不齐，有些甚至极为昂贵，使很多的中小型美业门店还在使用手工记账或在使用一些破解的非行业专业性的系统，致使这个行业信息化任然较为落后，生产力得不到提高，不能对店铺进行精准、高效的管理，员工需要花大量的时间做一些琐碎的，与核心业务无关的工作，增加了店铺的经营成本

1.3开发本系统的目的和意义

为了更好的帮助美业的发展，让更多美业的门店拥有简单实用的信息化管理工具，提高他们的生产力，让员工能够将更多的精力放在核心业务上，并降低店铺的运营成本，做到精准、高效的管理，是开发本系统的目的。该系统主要针对美容、美发和美甲门店处于初创阶段或者门店经营业绩暂时不是很高的门店，需要使用系统化的工具管理实现门店的收银、会员管理、会员充值、打折、员工业绩提成、店面营业报表统计等功能，以便老板能及时掌握客户的充值情况、消费情况和门店经营情况。该系统不仅可以把会员信息录入并进行信息化的管理，还可以对会员信息进行分析筛选，能够把会员的消费服务项目的数据流水作为店铺业务分析的工具，从中可以发现好的营销点和卖点，并能对不好的项目做出改善。老板能够通过该平台追踪每一笔经销存信息，对每一个员工的报酬、每个会员交易记录进行统一管理。

1.4本论文内容的主要安排

本论文主要是围绕海豚湾美甲店美业管理系统设计与开发而展开的。

第一章是绪论。阐述了我国信息化的发展现状，和我国企业信息化的发展现状，以及美业行业信息化的发展现状。然后介绍了本系统的开发背景，以及本系统开发的目的及意义，对以后的分析和开发起到了指导性作用。

第二章主要根据海豚湾美甲店的实际业务，对海豚湾美甲店美业管理系统进行了需求分析和可行性分析，以及根据该店铺的实际组织结构和业务流程绘制了系统组织结构图、业务流程图和数据流程图等。

第三章是在第二章所进行的系统分析的前提下，对要开发的系统做出了体系结构设计、代码设计、数据库设计以及输入输出设计。

第四章进入了系统开发的实施阶段。主要介绍了系统所用到的开发工具以及数据库的连接方式和过程。

第五章主要是系统的测试和维护。主要介绍了系统测试所用到的方法以及系统维护的内容和所需要的一些管理工作。

1. 海豚湾美甲店美业管理系统的分析

系统分析是设计开发管理信息系统的主要和关键阶段。

2.1系统需求分析

需求分析需要理清客户根据他们的日常业务需要些什么样的功能，使开发的系统既能满足他们的功能需求，也能满足他们日常运营的需求，还能通过所开发的系统，降低店铺的经营成本提高店铺的经营业绩。

2.1.1调查用户需求

本系统的最终用户为美甲店铺的管理员和店长。根据根据店长对日常业务的运营结合对美甲人群的调查，得出了下列实际需求。

1. 店铺的基本情况。

每一家店铺都会有一名或多名管理员以及若干店员和若干顾客。

1、店铺员工的基本信息。每一位员工入职时都会为其分配一个员工号，顾客结算买单进行员工提成时，就会与员工的员工号进行关联。以及顾客进行预约选择要预约的员工时也会员工的员工号进行关联。

2、预约的基本信息。当有店铺需要预约时就会生成一个预约编号。预约时顾客可以提出预约的开始时间和结束时间，并且可以选择要预约的员工，则该预约的预约信息就会和员工的员工号关联起来。预约会有散客和会员之分，会员预约可以选择常用员工。

3、会员的基本信息。当有顾客需要在该店办理成为会员时，就会生成会员信息。办理会员时顾客必须至少要办理一张会员卡。当会员下次到店里消费时，就可以凭拥有的会员卡享受相应地折扣或优惠。所有会员的会员号和店铺的会员卡的卡号之间会有关联关系。

4、店铺经营项目的基本信息。当店铺每上架一款经营项目时，就会产生了该项目的基本信息。项目供顾客购买，顾客在结算时就会选择所购买的项目进行结算，顾客和项目之间就会产生关联关系，以及在结算时又会选择该项目的提成员工，项目和员工之间也会产生一定的关联关系。

5、收银的基本信息。顾客消费完之后进行结算时就会产生收银信息。收银时会选择顾客所消费的经营项目，选择完经营项目后，若该位结算的顾客是该店铺的会员，则可以选择该会员所拥有的会员卡进行打折或优惠,最后在选择提成的员工。收银会将会员、会员卡和员工三者联系起来。

6、系统管理员的基本信息。当某家店铺需要采用本系统时，需要在该系统进行注册,注册时会录入注册用户的手机号和密码供他以后登录使用。当用户注册后系统会自动为该用户分配一个企业ID，即enterprise\_id。企业ID会是店铺其他所有业务关联起来的最重要的字段。

7、店铺的经营流水。店铺的经营流水记录了店铺每一笔收银，通过店铺的经营流水店长可以看到店铺每周、每天和每个月的经营业绩，可以一眼就看出哪一位员工的业绩最好提成最多，哪一位员工的业绩最差提成最少，哪一服务买得最好，哪一项服务卖得最差。通过这些一眼就能看出的数据，就可以对员工实行一些奖惩机制，对卖得较差的服务进行优化和巩固，实在不行可以下架该服务。

2.2系统可行性分析

2.2.1技术可行性分析

海豚湾美甲店美业管理系统是一个典型的B/S架构的系统。后端部分所采用的是NodeJS作为服务器端技术，MySQL作为数据库。前端部分用到了HTML、CSS和JavaScript等Web前端技术。

NodeJS即是服务器端版本的JavaScript，它相比传统客户端版本的JavaScript多了文件操作，I/O操作和网络服务的特性。NodeJS所提供的范式跟传统的Web服务器（如IIS和Apache等）不同，开发者写的程序就是Web服务器，NodeJS只是给开发者提供了一个构建Web服务器的框架。NodeJS有着强大的生态系统，所有的主流关系型数据库（MySQL、MariaDB、Oracle、SQL Server、PostgreSQL）的接口都有，不仅有NPM(Node Package Manager)作为NodeJS的包管理和分发工具，还有一个精简、灵活的Web框架——Express，为构建单页、多页及混合的Web程序提供了一系列健壮的功能特性。

NodeJS技术的特点：

1、异步I/O。程序在执行过程中必然要进行很多I/O操作，如读写文件、输入与输出、请求与相应等。由于I/O操作通常都很费时，在传统的编程模式中，如果碰到需要读取一个上千兆的文件的情况，整个线程都需要暂停下来用于读取文件，不能再执行其他程序，必须要直到文件读取完成后才能继续执行其他程序。换言之，I/O操作阻塞了代码的执行，极大的降低了程序效率。

2、事件循环与回调函数。事件循环是指NodeJS会把所有的异步操作使用事件机制来解决。基于事件的编程方式具有轻量级、松耦合、只关注事务等优点，并可通过NodeJS的async模块，使多个异步的任务同步执行。在NodeJS中，回调函数无处不在，回调函数是最好的接受异步调用返回数据的方式。这对开发者来说，意味着开发者必须知道有哪些事件，以及如何响应这些事件。在服务器上响应事件这种概念性的跳跃可能会比较难，但原理是一样的。

3、单线程。NodeJS保存了JavaScript在浏览器中单线程的特点。单线程的最大好处就是可以不像多线程那样处处在意状态的同步问题，没有死锁的存在，也没有线程上下文切换的开销。

4、跨平台。NodeJS是跨平台的，即同一套代码可以部署运行在Windows、Linux和OS X等平台。

MySQL是由Oracle公司运营的一款关系型数据库管理系统。它具有如下一些特点：

1. 支持多种字段类型。
2. 使用的核心线程是完全多线程，即支持多处理器。
3. MySQL可工作不同的平台上。支持Java、Python、PHP、Ruby和NodeJS等。
4. 全面支持SQL的GROUP BY和ORDER BY子句。
5. 可以在同一查询中混用来自不同的数据库的表。

NodeJS和MySQL都算是较为成熟的技术，应用也都比较广泛了，国内小有名气的网站酷厨（http://www.coochu.com）和花瓣（http://www.huaban.com）都是用NodeJS搭建的，所以开发本系统所采用的技术是可行的。

2.3系统组织结构图

根据海豚湾美甲店的日常运营及业务类型，可画出如图2-1所示的店铺的组织结构图。美甲店有一位店长，他主要负责店铺的运营工作，包括招聘美甲师，给美甲师发工资，顾客的收银、结算和会员卡的办理，对店铺的经营流水进行分析包括财务分析和业务分析，以及对店铺进行战略上的的规划等等。美甲师主要是为顾客提供服务，辅助店长对店铺的服务进行分析和优化。顾客主要是到店铺做服务的人，分为会员和散客，会员会拥有该店铺的会员卡，在收银结算的时候可以享受打折或减免的优惠，在会员生日的时候也可以享受相应地生日优惠活动，而散客什么也没有，只能按服务的原价支付。

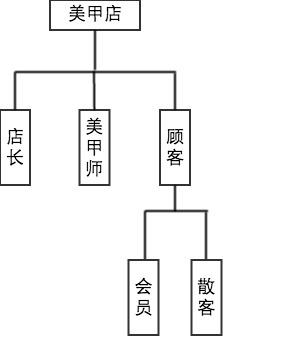


图 2-1海豚湾美甲店组织结构图

2.4系统业务流程图

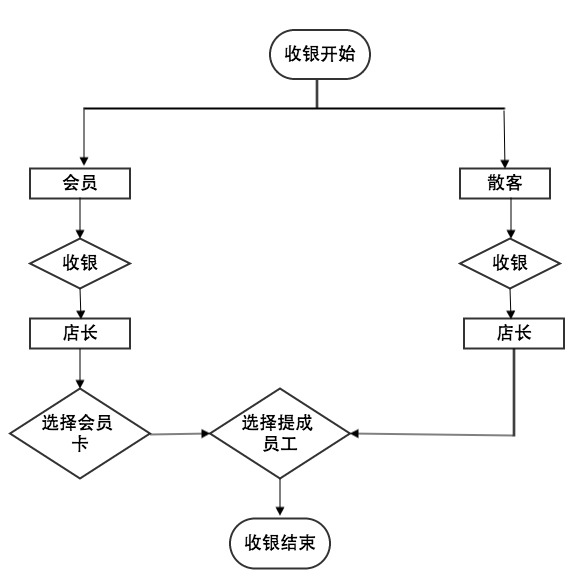


图 2-2店铺收银业务流程图

如图2-2所示为店铺的收银业务流程图，在店铺的收银业务流程中，如果结算的对象是会员则在结算的过程中可以选择该会员所拥有的会员卡给予相应地打折或减免，顾客再支付减免后的金额，然后再选择要提成的员工，就完成了该位顾客的收银。如果结算的顾客不是会员即散客，则在收银的过程中与会员收银的流程相比没有了选择会员卡的那一步，其余流程与会员收银的流程一样。

2.5系统数据流程图

数据流程图的主要用来反映功能需求和数据需求之间的联系，清晰的表达各模块之间数据的处理及加工，为之后的系统设计打下基础。

系统的顶层数据流图如图2-3所示。

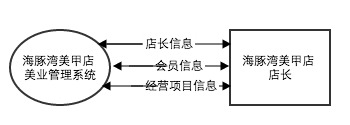


图 2-3系统顶层数据流图

2.6数据字典

数据字典为系统的数据库的创建提供字段依据和结果依据。根据数据字典可以知道需要创建哪些表，一张表中有哪些字段，哪些字段可以为空，哪些字段不可以为空，哪些字段必须唯一，哪些字段可以不唯一等。

海豚湾美甲店美业管理系统的数据字典设计如下：

|  |
| --- |
| 名字：管理员信息  别名：user  描述：系统管理员用来管理系统的员工信息、会员信息、会员卡信息和经营项目的信息  定义：user=id+username+password+signup\_time  位置：注册页面录入 |

|  |
| --- |
| 名字：员工信息  别名：staff  描述：用于对员工信息进行管理  定义：staff=id+staff\_name+staff\_phone+join\_time+enterprise\_id  位置：员工录入界面录入 |

|  |
| --- |
| 名字：会员信息  别名：member  描述：用于对会员的信息进行管理  定义：member=id+member\_name+member\_phone+join\_time+enterprise\_id  位置：会员录入界面录入 |

|  |
| --- |
| 名字：服务信息  别名：service  描述：用于对店铺服务的信息进行管理  定义：service=id+servce\_name+enterprise\_id  位置：经营项目录入界面录入 |

|  |
| --- |
| 名字：经营项目信息  别名：item  描述：用于对经营项目的信息进行管理  定义：item=id+item\_name+item\_price+service\_id+enterprise\_id  位置：经营项目录入界面录入 |

|  |
| --- |
| 名字：预约信息  别名：appointment  描述：用于对预约的信息进行管理  定义：appointment = id + staff\_name + start\_time + end\_time + member\_id + member\_name + enterprise\_id  位置：预约录入界面录入 |

|  |
| --- |
| 名字：会员卡信息  别名：card  描述：用于对会员卡的信息进行管理  定义：card=id+card\_name+card\_type+enterprise\_id  位置：会员卡录入界面录入 |

|  |
| --- |
| 名字：收银信息  别名：cash  描述：用于对顾客收银的信息进行管理  定义：cash=id+cash\_time+cash\_amount+cash\_staff+enterprise\_id  位置：收银录入界面录入 |

|  |
| --- |
| 名字：财务流水信息  别名：turnover  描述：用于查看店铺的经营流水信息  定义：turnover =id+item\_id+cash\_amount+enterprise\_id  位置：店铺管理台查看 |

1. 海豚湾美甲店美业管理系统的设计

3.1系统的体系结构

海豚湾美甲店美业管理系统的体系结构为B/S体系结构，即浏览器/服务器结构。基于这种结构，用户只需要有一台能够上网并且配备了网络浏览器（如谷歌的Chrome浏览器和微软的Internet Explorer浏览器等）的设备，向指定的域名或IP地址发起HTTP请求，Web服务器就会根据请求的URL给出相应的响应，用户就可以在浏览器中看到所需的页面和信息，若有数据信息存放在数据库，Web服务器就会向数据库发起操作取出相应的数据信息，若用户在在Web页面上进行了输入或点击操作，就会发出相应地HTTP请求，服务器也会相应处理这些请求并给出相应。

系统的主要功能有用户登录/注册功能、结算收银功能、顾客预约功能、会员管理功能、数据分析统计功能、服务及经营项目管理功能和会员卡管理功能。总体来说系统分为前台模块和后台模块。前台模块主要用于数据的展示，录入，对数据做一些格式化的处理以及向后台发送请求用于发送和接收数据。后台模块主要用于接收从前台发送的数据以及向前台发送数据，并对数据库进行相应地增加，删除，修改和查询。具体如图3-1所示：

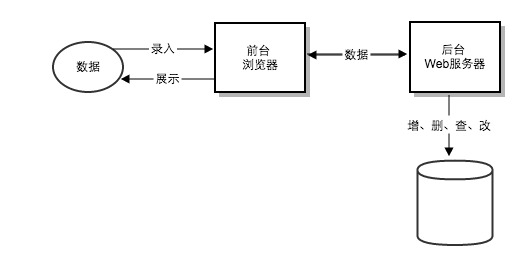


图 3-1系统体系结构图

3.1.1系统的前端

用户能通过前端界面看到店铺的经营流水，每个员工的业绩提成，店铺的预约情况，店铺会员的消费情况，每个项目才销售情况，同时也能对这些数据进行一定的分析统计，可以从中发现一些商业信息，从而能够帮助店主做一些经营决策，制定一些商业规划。

3.1.2系统的后端

系同的后端是整个系统运行的基础。系统的后端需要对整个系统做一定的权限控制，保证只有在用户登录后才能进行其他的操作。后台需要能够接收浏览器了的HTTP请求，以保证用户能访问系统，同时系统后台也需要接收从前台发送过来的数据，然后对数据做相应地处理和加工，再根据业务需求，通过SQL语句，对数据库做相应地操作。系统的后台也需要有一定的性能、安全性和稳定性的保证，尽可能地利用NodeJS异步执行的优势，对已不需要的数据库连接尽快的释放，还要做到防止SQL注入的危险。

3.2系统的代码设计

代码设计是开发管理信息系统的基础性工作。

海豚湾美甲店美业管理系统的代码设计采用了如下的编码方式。便于用户对系统进行识别和使用。

* 店铺管理员信息编号为字母“GL”,给店铺管理员录入的信息为“GL”加一位下划线再加三十二位的随机码就得到了店铺管理员的信息。例如，GL\_1061da40-f155-11e4-ae55-173d1f3c。
* 店铺会员信息编号为字母“HY”,给店铺会员录入的信息为“HY”加一位下划线再加三十二位的随机码就得到了店铺会员的信息。例如，HY\_1061da40-f155-11e4-ae55-173d1f3c。
* 店铺员工信息编号为字母“YG”,给店铺员工录入的信息为“YG”加一位下划线再加三十二位的随机码就得到了店铺YG的信息。例如，YG\_1061da40-f155-11e4-ae55-173d1f3c。
* 店铺服务信息编号为字母“FW”,给店铺的服务录入的信息为“FW”加一位下划线再加三十二位的随机码就得到了店铺服务的信息。例如，FW\_1061da40-f155-11e4-ae55-173d1f3c。
* 店铺经营项目信息编号为字母“HM”,给店铺经营项目录入的信息为“HM”加一位下划线再加三十二位的随机码就得到了店铺经营项目的信息。例如，HY\_1061da40-f155-11e4-ae55-173d1f3c。
* 店铺开设的会员卡信息编号为字母“HYK”,给店铺开设的会员卡录入的信息为“HYK”加一位下划线再加三十二位的随机码就得到了店铺所开设的会员卡的信息。例如，HYK\_1061da40-f155-11e4-ae55-173d1f3c。
* 店铺对每一笔收银的信息编号为字母“SY”,给店铺的每一笔收银录入的信息为“SY”加一位下划线再加三十二位的随机码就得到了店铺每一笔收银的信息。例如，SY\_1061da40-f155-11e4-ae55-173d1f3c。
* 店铺顾客预约的信息编号为字母“YY”,给店铺的每一位顾客进行预约时录入的信息为“YY”加一位下划线再加三十二位的随机码就得到了店铺每一位顾客预约的信息。例如，YY\_1061da40-f155-11e4-ae55-173d1f3c。

3.3系统的数据库设计

数据库设计主要是根据需求分析得出的数据字典中的字段及约束关系在数据库中进行逻辑设计，即将数据的概念模型转换成特定的数据库管理系统（DBMS，本系统即MySQL）所支持的数据模型。

表3-1店铺会员信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 会员编号 | varchar | 64 | 否 |
| username | 用户名 | varchar | 64 | 否 |
| mobilephone | 手机号 | varchar | 64 | 否 |
| join\_date | 入会时间 | double | 16 | 否 |
| birthday | 会员生日 | double | 16 | 是 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

表3-2管理员信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 店铺编号 | varchar | 64 | 否 |
| username | 管理员用户名 | varchar | 64 | 否 |
| password | 登录密码 | varchar | 64 | 否 |
| signup\_time | 注册时间 | double | 16 | 是 |

表3-3店铺所开设的会员卡信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 会员卡编号 | varchar | 64 | 否 |
| card\_name | 会员卡名称 | varchar | 32 | 否 |
| card\_type | 会员卡类型 | varchar | 16 | 否 |
| discount | 折扣量 | int | 16 | 是 |
| count | 次数 | int | 16 | 是 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

表3-4店铺服务分类信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 服务编号 | varchar | 64 | 否 |
| service\_name | 服务名称 | varchar | 32 | 否 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

表3-5店铺经营项目信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 项目编号 | varchar | 64 | 否 |
| item\_name | 项目名称 | varchar | 16 | 否 |
| item\_price | 项目价格 | float | 16 | 否 |
| service\_id | 服务编号 | varchar | 64 | 否 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

表3-6店铺预约信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 预约编号 | varchar | 64 | 否 |
| customer\_name | 顾客姓名 | varchar | 32 | 否 |
| customer\_phone | 顾客电话 | varchar | 32 | 否 |
| start\_time | 开始时间 | double | 16 | 否 |
| end\_time | 结束时间 | double | 16 | 否 |
| comment | 备注 | varchar | 128 | 是 |
| staff\_id | 预约员工 | varchar | 64 | 是 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

表3-7店铺员工信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 员工编号 | varchar | 64 | 否 |
| staff\_name | 员工姓名 | varchar | 32 | 否 |
| staff\_phone | 员工电话 | varchar | 16 | 否 |
| entry\_time | 入职时间 | double | 16 | 否 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

表3-8员工项目提成表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 提成编号 | varchar | 64 | 否 |
| tc\_staff\_id | 提成员工编号 | varchar | 64 | 否 |
| tc\_staff\_name | 提成员工姓名 | varchar | 32 | 是 |
| tc\_item\_id | 提成项目编号 | varchar | 64 | 否 |
| tc\_item\_name | 提成项目名称 | varchar | 32 | 是 |
| tc\_time | 提成时间 | double | 16 | 否 |
| tc\_amount | 提成金额 | float | 16 | 否 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

表3-9店铺收银信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 数据类型 | 长度 | 允许空 |
| id | 收银流水编号 | varchar | 64 | 否 |
| member\_id | 会员编号 | varchar | 64 | 是 |
| member\_name | 会员姓名 | varchar | 32 | 是 |
| card\_id | 会员卡编号 | varchar | 64 | 是 |
| card\_type | 会员卡类型 | varchar | 16 | 是 |
| pay\_money | 支付金额 | float | 16 | 否 |
| staff\_id | 提成员工编号 | varchar | 64 | 是 |
| enterprise\_id | 店铺编号 | varchar | 64 | 否 |

3.4系统的输入输出设计

输入/输出设计对用户尤为重要。它是一个组织系统形象的具体实现。

3.4.1系统的输入设计

海豚湾美甲店美业管理系统需要输入的信息主要包括店长用户名和密码的输入，会员资料的录入以及在收银结算时结算数据的录入。这些信息在录入的过程中均有很完备的输入校验防止用户乱输入信息，以及在输入的过程中也有很明确地文字提示，这些输入操作基本都能通过我们所熟悉的键盘或鼠标来完成，大大方便了用户的输入体验。

如题3-2所示，为系统添加会员时的输入界面，该界面尽量做到让用户少用键盘输入，能通过鼠标点击选择的就通过鼠标操作，如生日和性别等输入项。该界面也有一定的输入校验功能，例如在手机号码输入项，若用户输入的内容不是十一位的手机号码，用户就不会被允许的提交所输入的东西，并告知用户要输入全是数字的手机号。



图 3-2添加会员的输入界面

3.4.2系统的输出设计

如图3-3所示为顾客预约输出界面。该界面采用类似日历的表格形式来展示信息，让预约信息显得直观精准，用户一看就知道几月几号几点有哪位顾客预约了。能够大大提升用户体验。



图3-3顾客预约输出界面

1. 海豚湾美甲店美业管理系统的实施

4.1系统开发工具的选择

系统开发工具的选择决定了系统的开发成本和效率。

4.1.1MySQL数据库的选择

MySQL数据库是当前数据库领域最为流行的关系型数据库之一，它具有如下特点：

* 可以运行在不同的平台上。
* 可以在同一查询中混用来自不同数据库的表。
* 开源、免费能降低开发成本。
* 社区活跃。碰到问题能快速找到解决方案。
* 执行速度较快，能提高系统的性能。
* 学习资料较多，入门容易。

4.1.2NodeJS的选择

NodeJS主要用于开发系统的后端，主要用于接收和响应HTTP请求以及操作数据库。NodeJS是开发该系统的关键技术。

NodeJS具有如下特点：

* 它是一个JavaScript运行环境。
* 依赖于Chrome V8引擎进行代码解释。
* 事件驱动。
* 非阻塞I/O。
* 轻量、可伸缩，适用于数据交互应用。
* 单线程，单进程。

根据NodeJS的特点可知，NodeJS有利于开发高性能的应用。这也是开发本系统所期望的。

4.1.3WebStorm的选择

WebStorm是最流行的NodeJS集成开发工具（IDE），由Jetbrains公司发行和维护。是本系统的主要开发工具和调试工具。WebStorm具有如下特点：

1、支持自动保存。不需要开发者一次又一次地手动保存啦，所有的操作都直接存储，当然，万一键盘误操作也会被立即存储，不过我们可以通过本地版本控制解决这个问题。

2、任何一个编辑器只要文件关闭了就没有历史记录了，但是WebStorm有。就是说，只要WebWtorm不关闭，你的文件随时可以返回到之前的操作，WebStorm关闭重启后这些历史记录就没有了。这样的坏处也是显然的，由此带来的内存消耗也必然比较大。

3、任何一个编辑器，除了服务器svn之外，没有本地版本，但是WebStorm提供一个本地文件修改历史记录。

4、与时俱进的眼光。Zencoding于2009年出现于IT界，在这之后，鲜有工具直接集成到里边。WebStorm 2.0之后就集成了。NodeJS,HTML5,Git,CVS等 就不一一列举了。

5、提供的插件也是比较齐全，安装非常方便。这样带来了另外的问题，以前的Eclipse是安装第三方的，WebStorm貌似不能安装第三方的插件。

6、可以导出当前设置：File -> Export setting。

4.1.4Git的选择

在开发本系统时用到了Git对代码进行版本控制。用源码控制系统有很多原因，但对我来说最大的收益还是它的根本属性：知道什么时候谁做过什么修改，这样在有必要时我就可以询问更多信息。要了解项目的历史，版本控制是最好的工具之一。传统的集中式版本控制系统不能有效地管理分支和进行分支间合并。集中管理的版本库只有唯一的分支命名空间，需要专人管理，从而造成分支创建的不自由；分支间的合并要么因为缺乏追踪导致重复合并、引发严重冲突，要么因为版本控制系统本身蹩脚的设计导致分支合并时效率低下和陷阱重重。Git凭借其灵活的设计让项目摆脱分支管理的梦魇。

4.1.5Bootstrap的选择

Boostrap是最流行的前端框架，很多个人网站以及一些公司的内部系统都采用了Bootstrap。利用Bootstrap可以快速开发出美观、适用、响应式的和交互友好的前端。

本系统采用Bootstrap主要想利用Bootstrap的响应式布局，模态框以及美观漂亮的CSS组件和JavaScript组件。

4.1.5AngularJS的选择

AngularJS󰀃是一个为动态WEB应用设计的结构框架。它使用HTML作为模板语言，通过扩展HTML的语法，让开发者能更清楚、简洁地构建应用组件。它的创新点在于，利用数据绑定和依赖注入，它使开发者不用再写大量的代码了。这些全都是通过浏览器端的Javascript实现，这也使得它能够完美地和任何服务器端技术结合。

本系统采用AngularJS主要想利用AngularJS的数据绑定，数据过滤以及用来发送HTTP请求。

4.2数据库的连接

本系统用到了连接池来处理程序对数据库的访问和操作。创建数据库连接主要分三步进行：第一是创建用于保存数据库配置信息的文件dbConnection.js;第二是创建一个用于数据库连接和执行SQL语句的公共组件dbHelper.js；第三是在需要进行数据库操作的程序中引入dbHelper.js文件（vardbHelper = require(‘dbHelper.js’)）。下面将通过代码进行详细介绍。

本系统的数据库连接的配置信息保存在dbConnection.js文件中,要连接数据库就必须要引人这个文件。它指明了数据库的地址、名称、用户名和密码，并创建了连接池pool作为导出对象。代码如下：

varmysql = require('mysql');

var options = {

host: 'localhost',

port: '3306',

database: 'graduationProject',

user: 'root',

password: ''

};

exports.pool = mysql.createPool(options);

数据库连接文件dbHelper.js用于从连接池中取出一个连接并对数据库进行操作，然后再释放连接。代码如下：

vardbConnection = require('../config/dbConnection.js');

var pool = dbConnection.pool;

exports.execSql = execSql;

functionexecSql(sql, option, callback){

pool.getConnection(function(err, connection){

if(err){

console.log('与MySQL数据库建立连接失败。');

callback(err);

}

else{

connection.query(sql, option, function(err, rows){

if(err){

console.log('数据操作失败');

callback(err);

}

else{

callback(null,rows);

connection.release();

}

});

}

});

}

在需要进行数据库操作的程序中，必须引入dbHelper.js文件，当程序需要执行SQL语句时，即可调用dbHelper.js中的execSql()方法，对数据库进行增，删，查，改（CRUD）操作。

4.3系统的详细设计及实现

4.3.1系统的登录和注册模块的设计及实现

在用户登录界面如图4-1所示，用户需要在登录表单中填写在注册时所填写的用户名和密码才能进入系统。该登录表单有一定的验证机制，若用户表单中有某一字段未填写就提交表单，会有相应的提示信息告知用户填写该字段如图4-2所示。当用户的用户名或密码输错后也会有相应的提示信息如图4-3、4-4所示。如果用户没有账户，则需要先注册一个账户，就可以点击“注册一个账号”跳转到注册页面如图4-5所示。注册页面的表单也有和登录页面的表单一样的不允许提交空字段的验证。



图 4-4用户登录界面



图 4-5有字段未填写的提示信息



图 4-6密码错误的提示信息



图7-4用户名填写错误的提示信息



图 4-8用户注册页面

4.3.2系统的收银模块的设计及实现

当顾客需要买单时，店员会进入收银界面，若该顾客是会员，选择该顾客的会员号及该顾客所做的服务会出现图4-6所示的界面，该界面会显示该顾客所做各个服务的价格及所有服务的总价格。

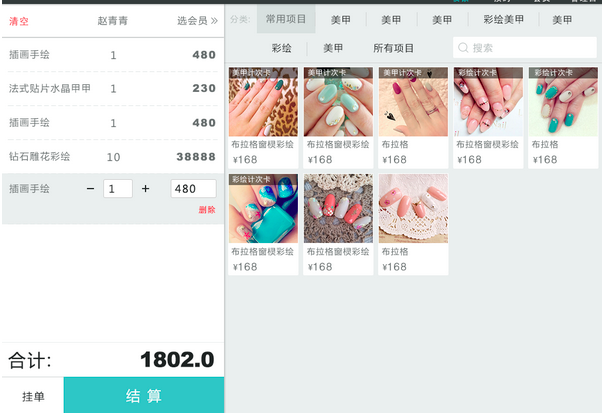


图 4-6会员收银界面

点击图4-6界面中的“结算”后，会出现图4-7所示的会员收银结算界面，在该界面中可以选择结算顾客所拥有的打折卡或其他卡券进行结算，然后再选择现金或银行卡对剩余的金额进行支付。



图 4-7会员收银的结算界面

点击图4-7界面中的“下一步”后，会出现图4-8所示的会员收银提成界面，在该界面中可以选择对该顾客做过服务的员工进行提成，点击“完成”之后即完成了本次收银。



图 4-8收银提成界面

散客收银的界面和设计与会员收银的界面和设计基本一样，只是没有收银最开始的会员选择过程和结算时选优惠卡的过程。

4.3.3系统的预约模块的设计及实现

如果有顾客需要提前预约某位美甲师在某个时间进行服务，则可进入会员预约界面如图4-9所示，该界面会显示那个时间那个美甲师被预约了，点击右上角的“添加”，就会呈现出入图4-10所示的添加预约的界面。



图 4-9预约界面

在添加预约界面，默认是散客进行预约，如果会员要进行预约则需要先选择会员。选择了要预约的员工及预约的时间点后，点击右上角的“完成”，即完成了本次预约。



图 4-10添加预约界面

成功预约后，预约的缩略信息就会出现在图4-9所示的预约界面的时间表中，点击该缩略信息就可以看到该预约的详细信息。

4.3.4系统的会员管理模块的设计及实现

一个美甲店铺，它会有很多会员。需要对这些会员进行高效的记录和管理，记录他们所拥有的卡券和消费明细，以便做到对店铺的精细化管理。比如能够知道什么样的会员卡最受会员欢迎，从他们的消费明细中可以知道他们最喜欢做哪些服务以及喜欢哪位美甲师为他们提供服务。

会员界面如图4-11所示，在该界面中的搜索框内，只需输入用户的手机号，即可找到该会员，可看到该会员的入会时间、姓名、生日以及所拥有的卡券等信息，点击右上角的“添加”即可添加一位会员。



图 4-11

当点击会员列表的某位会员后，就会进入该会员的详细资料界面如图4-12所示。在该界面用户不仅可以看到他所有卡券的详细信息，也可以看到他的消费明细，还可以通过点击添加卡项办理一张新的会员卡。



图 4-12

1. 海豚湾美甲店美业管理系统的测试与维护

5.1系统测试

一个没有经过测试的系统是不能上线的。系统测试是系统上线后能够稳定、正确和高效运行的重要保障。系统测试需要对系统的用户界面（UI）、交互、功能、性能和稳定性进行测试。

在系统测试过程中要仔细浏览检查系统的每个业务逻辑的每一个用户界面，若发现有样式错乱或交互不友好的地方要及时的修复改正。在系统测试过程中需要根据用户提出的需求，检查系统是否有相应地功能，若没有要及时添加相应地功能，若有相应的功能，则需要通过输入一些真实地数据，运行一下该功能，观察运行过程中是否会出错，输入的数据是否能进入相应的数据库表，在需要查询的地方是否可以查到所需的数据，在数据需要进行加工处理的地方，是否进行了正确地加工处理，在每一步当中若发现了错误都要在不影响其他功能的前提下修复改正。随着用户的增多，系统的并发数量越来越高，这就需要系统有一定的性能保障，在数据库连接方面要用连接池，连接用完后要及时释放连接，在涉及到I/O操作时，要充分利用NodeJS异步执行的优势，减少等待。由于NodeJS是单进程，执行一遇到错误就会造成整个进程的崩溃，所以系统需要有一定的容错机制，仿止系统崩溃，提高稳定性。

5.2系统维护

系统维护是系统能够持续正常运行和系统使用寿命的重要保障，系统维护是一个长期的过程，伴随着系统的整个生命周期。

系统在运行的过程中，一旦发生了运行故障或异常时，造成系统运行异常，不能完成特定任务时，就需要对故障或异常进行跟踪定位，找出造成异常或故障的原因，并修复造成该异常的Bug，避免再次发生同样地故障或异常。

当系统的运行环境发生变化时，如硬件变化或运行的软件环境发生了变化，系统需要稳定的迁移到新环境中，能够在新环境中稳定、正确的运行，这可能需要对系统的源码做一些修改和重新配置。这需要系统有较好的可维护性，能够不断适应外部运行环境的变化。

当客户又有新的需求提出时，需要开发新的功能或者调整已有的功能，这就需要系统有较好的可扩展性，能够在原有系统的基础上不断地开发新功能或对已有的功能进行调整，新功能或调整后的功能又能够和系统原有的功能配合起来使用。

系统维护不仅需要系统维护人员有一定的需求分析，总结和归纳的能力，还需要有一定的故障跟踪，定位和检查能力，以及开发调试能力。

结 论

经过对本次毕业设计论文的编写以及系统的开发，让我认识到管理信息系统的开发是一个系统性的工程。它涉及到对一个行业的观察、调研，对用户需求的收集，分析和整理，对项目的可行性进行分析和评估，需要将实际的业务模型转换成用于分析的逻辑模型，在开发的阶段又需要将分析时的逻辑模型转换为物理模型，在开发的过程中又会随时面临需求的调整，系统开发完成之后还需要进行实施，测试和维护。每一个过程都需要具备扎实的专业知识，还需要耗费大量的时间和精力。因此，通过这次毕业设计，让我对管理信息系统的开发不仅有了切身的体会，更有了进一步的认识，还让我巩固了在信息管理与信息系统专业四年所学的各项知识，同时也坚定了我继续投身信息系统的学习、研究与开发的决心与信心。

本系统所采用的主要技术NodeJS相比于Java、Python和PHP等算是比较新的技术，对NodeJS理解得还不是很彻底，一开始很不习惯这种异步编程的风格，导致造成了很多Bug，我也是一边学习一边开发，深知整个程序的隐患和不足之处，由于本系统是基于谷歌的Chrome浏览器开发的，没有在其它浏览器上测试过，也没有做相应地适配，在其它浏览器上很可能存在一些兼容性的问题，比如样式错乱，需要在后续的维护中不断的完善，修复和扩展。本系统由于条件所限，比如由于几乎没有用户基数，无法进行真实地高并发测试，所以还没有达到一个可以上线的商业软件的水准，仅可供交流和学习使用，另外本系统所有的源代码都托管在Github（http://www.github.com）上面进行开源，所有人都可以下载使用。