**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

20世纪90年代以来，信息化以及信息技术所推动的技术革命，其必要性和重要性已经被世人所认同，不论人们的主观意愿如何，信息化必将渗透到各行各业，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。

企业信息化在我国信息化领域是起步较早的部分，尤其是大中型企业。包括金融、电信、汽车等行业的信息化已经取得了阶段性成果并逐渐步入了成熟阶段。主要表现在：这些企业大都设立了信息化管理机构，企业基础网络及市话建设基本完成，财务管理系统和办公自动化建设上了新台阶。中小企业是国民经济中最具活力的组成部分，近几年来，中小企业的信息化意识也有了一定提高，但发展水平参差不齐。

2014年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基于互联网环境的信息化下的用户研究，在传统企业如何落地应用。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

系统分析是管理信息系统的主要和关键阶段。系统分析把要解决的问题作为一个系统，对系统要素进行综合分析，找出解决问题的可行方案。系统分析的主要任务是将在系统详细调查中所得到的文档资料集中到一起，对组织内部整体管理状况和信息处理过程进行分析。

2.1系统需求分析

需求分析要对解决的问题进行详细的分析，弄清楚问题的要求，包括需要输入什么数据，要得到什么结果以及最后应输出什么。

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。

2.2系统可行性分析

可行性分析的目的是用最小的代价在尽可能短的时间内确定问题是否能够解决。其实质上是要进行一次大大压缩和简化系统的分析和设计过程，也就是在较高层次上以比较抽象的方式进行的系统分析和设计的过程。我所设计的海豚湾美甲店美业管理系统是一个规模不太大的系统，但是目标明确，清晰的描述了对目标系统的一切限制和约束。

2.2．1技术可行性分析

本系统是一个典型的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是服务器端的JavaScript运行环境，它具有无阻塞（non-blocking）和事件驱动（event-driven）等特色，NodeJS采用V8引擎，实现了类似Apache和Nginx的Web服务，让开发者可以通过它来搭建基于JavaScript的Web应用。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）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，要读一个几G的文件，需要整个线程都暂停下来，等待文件读完后继续执行。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，有个线程在不断地循环检测事件队列。事件循环会检查事件队列中有没有未处理的事件，直到程序结束。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。

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数据库的表。

综上所述，开发本系统所采用的技术都已比较成熟，可以完成系统的开发。

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。

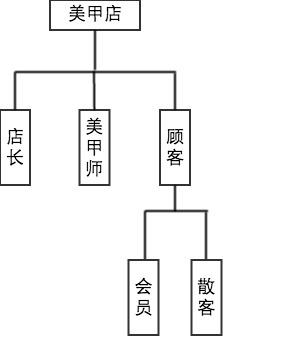


图 2-1海豚湾美甲店组织结构图

2.4系统业务流程图

业务流程分析可以帮助了解某项业务的具体处理过程，发现和处理系统调查工作中的错误和瑕疵，修改和删除原系统不合理的部分，在新系统的基础上优化处理流程。业务流程分析是对业务功能分析的进一步细化，从而得到业务流程图即TFD （Transaction Flow Diagram ）,业务流程图用一些尽可能少的规定的符号及连线来表示某个具体业务处理过程，业务流程图易于阅读和理解，是分析业务流程的重要步骤。

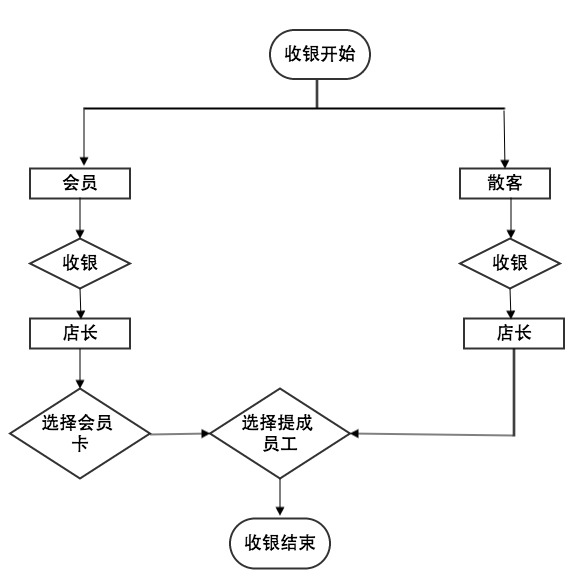


图 2-2店铺收银业务流程图

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。

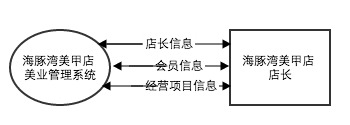


图 2-3系统顶层数据流图

2.6数据字典

数据字典则是系统中各类数据描述的集合,是进行详细的数据收集和数据分析所获得的主要成果。数据字典通常包括数据项、数据结构、数据流、数据存储和处理过程五个部分。数据字典是关于数据的信息的集合，也就是对数据流图中包含的所有元素的定义的集合。数据字典存放有数据库所用的有关信息，对用户来说是一组只读的表。数据字典的内容包括：数据库中所有模式对象的信息，如表、视图、簇、及索引等。分配多少空间，当前使用了多少空间等。列的缺省值。 约束信息的完整性。用户的名字。用户及角色被授予的权限。用户访问或使用的审计信息。其它产生的数据库信息。

编写数据字典是系统开发的一项重要的基础工作。一旦建立，并按编号排序之后，就是一本可供查阅的关于数据的字典，从系统分析一直到系统设计和实施都要使用它。在数据字典的建立、修正和补充过程中，始终要保证数据的一致性和完整性。

该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 |

1. 海豚湾美甲店美业管理系统的设计

系统设计根据系统分析阶段所确定的新系统的逻辑模型、功能要求，在用户提供的环境条件下，设计出一个能在计算机网络环境上实施的方案，即建立新系统的物理模型。

3.1系统的体系结构

本系统的体系结构为B/S（Browser/Server,浏览器/服务器模式）结构。B/S结构的最大特点是：用户可以通过网络浏览器（如Chrome、Internet Explorer等）去访问Internet上的文本、数据、图像、动画、视频和声音等信息，这些信息都是由许许多多的Web服务器产生的，而每一个Web服务器又可以通过各种方式与数据库服务器连接，大量的数据实际存放在数据库服务器中。客户端除了浏览器，一般无须任何其他应用程序，用户只需通过浏览器向Web服务器发起HTTP请求,Web服务器再给出相应地响应，用户就可以在浏览器中看到所需的信息，若有数据信息存放在数据库，则Web服务器的程序中就会有操作数据库有关的指令，由Web服务器交给数据库服务器来解释执行，并返回给Web服务器，Web服务器又返回给用户。在这种结构中，最大的好处就是运行维护比较简便，能够实现不同的人员，从不同的地点，以不同的接入方式（比如LAN, WAN, Internet/Intranet等）访问和操作共同的数据；最大的缺点是对企业外网环境的依赖性太强，由于各种原因引起企业外网中断都会造成系统瘫痪。

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：

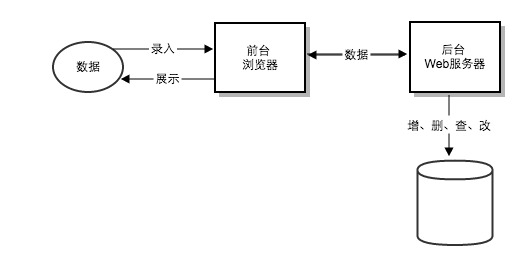


图 3-1系统体系结构图

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。代码设计的主要工作是把信息系统中所涉及到的信息的某种符合体系转换成便于计算机识别和处理的另一种符合体系的过程。代码设计的目的是要为信息系统设计出一套优化的代码方案，提交给用户使用和维护。

代码设计一般可分为两步：第一步是编制系统的编码分类体系表，用它来从总体上把握信息分类的情况，展示编码对象的内部联系；第二步是对信息具体编码，根据编码分类体系表所给出的编码对象，选择合适的代码结构，确定每一个编码对象的码值。

本系统的代码设计的内部信息编码采用了统一的编码方式。

* 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。

3.3系统的数据库设计

数据库设计主要是进行数据库的逻辑设计，即将概念数据模型转换成特定的数据库管理系统（DBMS）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；它能够为用户建立良好的工作环境，激发用户主动工作、努力学习的热情；它只要符合用户习惯，方便用户操作，就能使目标使目标系统易于为用户所接受；它为用户提供了易读易懂的信息形态。

3.4.1系统的输入设计

输入设计是信息系统与用户之间交互的纽带，设计的任务是根据具体业务的具体要求，确定适当的输入形式，使信息系统获取管理工作中所产生的正确的信息。输入设计的目的是提高输入效率，减少输入错误。输入设计应做到控制输入量，尽可能利用计算；减少输入延迟，尽量批量输入、周转文件输入；减少输入错误，采用多种校验方法和验证技术；避免额外多余的步骤，简化输入过程。

本系统输入信息主要包括会员资料的录入,店长用户名和密码的输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添加会员输入界面

3.4.2系统的输出设计

输出设计的任务是使目标系统的输出的信息是能够满足用户需求的信息。输出设计的目的是为了正确及时地反映和组成用于管理各部门所需要的信息。信息能否满足用户的需要，直接关系到系统的使用效果和系统的成功与否。系统设计人员应该给出系统输出的说明，这个说明既是将来编程人员在软件开发中进行实际输出设计的依据，也是用户评价系统实用性的依据。因此，设计人员要能选择合适的输出方法，并以清楚的方式表达出来。输出的方法主要有表格、图形和图标。

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

在信息系统分析与设计的阶段中，开发人员为新系统设计了它的逻辑模型和物理模型。系统实施阶段的目标就是把系统设计的物理模型转换成可实际运行的新系统。系统实施阶段既是系统的成功实现的最后一步，又是取得用户对新系统信任的关键阶段。

4.1系统开发工具的选择

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：

* 可以同时处理几乎不限数量的用户。
* 处理多大50,000,000以上的记录。
* 命令执行速度快，也许是现今最快的。
* 简单有效的用户特权体系。
* 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。

4.1.2NodeJS的选择

NodeJS是一个基于Chrome JavaScript运行时建立的平台，是一个JavaScript的运行环境(runtime)。实际上它是对Google V8引擎的封装。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。

JavaScript是一个事件驱动语言，NodeJS利用了这个优点，编写出可扩展性高的服务器。NodeJS采用“事件循环”(event loop)的架构,使得编写可扩展性高的服务器变得既容易又安全。NodeJS采用一系列“非阻塞”库来支持事件循环的方式。本质上就是为文件系统、数据库之类的资源提供接口。向文件系统发送一个请求时，无需等待硬盘(寻址并检索文件)，硬盘准备好的时候非阻塞接口会通知NodeJS。该模型以可扩展的方式简化了对慢资源的访问。有利于开发高性能的应用。

4.1.3WebStorm的选择

WebStorm 是jetbrains公司旗下一款JavaScript 开发工具。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。

Git是一款免费、开源的分布式版本控制系统，最早由Linus Torvalds创建，用于管理Linux内核开发，现已成为分布式版本控制的主流工具。版本控制系统不但要忠实地记录数据的每一次变更，还要能够帮助还原任何一次历史变更，以及实现团队的协同工作等。版本控制能决定项目的成败，甚至是公司的生死，此言不虚。我在推广开源项目管理工具和为企业提供咨询服务的过程中看到，有很多团队因为版本控制系统管理的混乱导致项目延期、修正的 Bug 重现、客户的问题不能在代码中定位……无论他们使用的是什么版本控制系统（开源的或是商业的）都是如此。这是因为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

Bootstrap是著名的社交网站、微博的先驱Twitter在2011年8月推出的开源WEB前端框架，集合CSS和HTML，使用了最新的浏览器技术，为快速WEB开发提供了一套前端工具包，包括布局、网格、表格、按钮、表单、导航、提示等等。使用Bootstrap可以构建出非常优雅的前端界面，而且占用资源非常小。

Bootstrap发布时间不长，但是已经非常成熟了，目前最新版本是3.3.4，包括完整的CSS编译和非编译版本，样例模板和Javascript插件。Bootstrap中包含了丰富的Web组件，根据这些组件，可以快速的搭建一个漂亮、功能完备的网站。其中包括以下组件：

下拉菜单、按钮组、按钮下拉菜单、导航、导航条、面包屑、分页、排版、缩略图、警告对话框、进度条、媒体对象等。

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。

AngularJS是为了克服HTML在构建应用上的不足而设计的。HTML是一门很好的为静态文本展示设计的声明式语言，但要构建WEB应用的话它就显得乏力了。所以AngularJS做了一些工作来让浏览器做开发者想要的事。通常，我们是通过以下技术来解决静态网页技术在构建动态应用上的不足：

类库--类库是一些函数的集合，它能帮助开发者写WEB应用。起主导作用的是开发者的代码，由开发者来决定何时使用类库。类库有：jQuery等

框架--框架是一种特殊的、已经实现了的WEB应用，开发者只需要对它填充具体的业务逻辑。这里框架是起主导作用的，由它来根据具体的应用逻辑来调用你的代码。框架有：knockout、sproutcore等。

AngularJS使用了不同的方法，它尝试去补足HTML本身在构建应用方面的缺陷。AngularJS通过使用我们称为标识符(directives)的结构，让浏览器能够识别新的语法。例如：使用双大括号{{}}语法进行数据绑定；使用DOM控制结构来实现迭代或者隐藏DOM片段；支持表单和表单的验证；能将逻辑代码关联到相关的DOM元素上；能将HTML分组成可重用的组件。

AngularJS试图成为成为WEB应用中的一种端对端的解决方案。这意味着它不只是开发者的WEB应用中的一个小部分，而是一个完整的端对端的解决方案。这会让AngularJS在构建一个CRUD（增加Create、查询Retrieve、更新Update、删除Delete）的应用时显得很“固执”。但是，尽管它很“固执”，它仍然能确保它的“固执”只是在开发者构建应用的起点，并且开发者仍能灵活变动。 AngularJS的一些出众之处如下：

构建一个CRUD应用可能用到的全部内容包括：数据绑定、基本模板标识符、表单验证、路由、深度链接、组件重用、依赖注入。

测试方面包括：单元测试、端对端测试、模拟和自动化测试框架。具有目录布局和测试脚本的种子应用作为起点。

AngularJS通过为开发者呈现一个更高层次的抽象来简化应用的开发。如同其他的抽象技术一样，这也会损失一部分灵活性。换句话说，并不是所有的应用都适合用AngularJS来做。AngularJS主要考虑的是构建CRUD应用。幸运的是，至少90%的WEB应用都是CRUD应用。但是要了解什么适合用AngularJS构建，就得了解什么不适合用AngularJS构建。如游戏，图形界面编辑器，这种DOM操作很频繁也很复杂的应用，和CRUD应用就有很大的不同，它们不适合用AngularJS来构建。像这种情况用一些更轻量、简单的技术如jQuery可能会更好。

4.2数据库的连接

在进行系统开发时，一个很重要的步骤就是建立程序到数据库的连接，用以访问数据库。连接数据库有DSN(Data Source Name)和非DSN两种方法。应用DSN访问数据库需要配置ODBC数据源（即系统DSN），该方法的优点是比较安全；而应用非DSN访问数据库不需要配置ODBC数据源，比较方便，但不够安全。本系统采用DSN的方式访问数据库。

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：一是创建数据库配置文件dbConnection.js;二是创建数据库连接文件dbHelper.js；三是在需要进行数据库操作的程序中引入dbHelper.js文件（var dbHelper = require(‘dbHelper.js’)）。下面进行详细介绍。

本系统的数据库的配置文件为dbConnection.js,对数据库操作是必不可少的。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：

var mysql = 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：

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

var pool = dbConnection.pool;

exports.execSql = execSql;

function execSql(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。

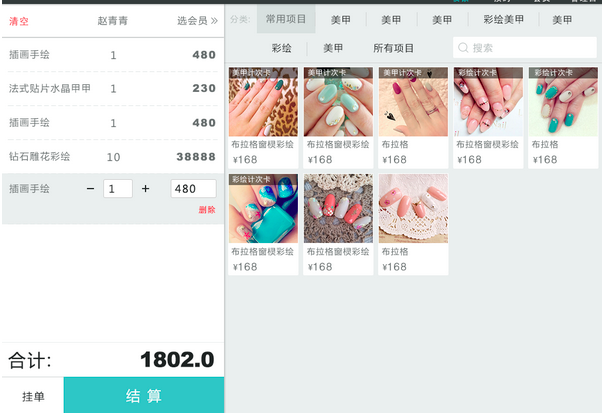


图 4-6会员收银界面

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

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

系统测试是系统开发过程的重要组成部分，是用来确认一个系统的品质或性能是否符合开发之前所提出的一些要求。系统测试的目的，第一是要确认系统的质量，其一方面是确认系统做了我们所期望的事情（Do the right thing），另一方面是确认系统以正确的方式来做了这个事（Do it right）；第二是提供信息，比如提供给开发人员或产品经理的反馈信息，可作为风险评估的信息；第三系统测试不仅是在测试系统的本身，而且还包括系统开发的过程。如果一个系统开发完成之后出现了很多问题，这说明此系统开发过程很可能是有缺陷的。

5.1.1系统测试的原则

在进行正式测试之前，测试人员应该知道系统测试的一些基本原则。这里有一组测试原则：

1、测试应从“小规模”开始，逐步转向“大规模”。

2、穷尽测试是不可能的。

3、不充分测试是不负责任的；过分测试是一种资源的浪费，同样也是一种不负责任的表现。

4、为了达到最佳效果，应该又独立的第三方来测试。

5、所有的测试都应该追溯到用户需求。

6、应该在测试工作真正开始前的较长时间内就进行测试计划。

7、一定要注意测试中的错误集中发生的现象。

8、回归测试的关联性一定要引起充分的注意，修改一个错误而引起更多的错误是很常见的现象。

5.1.1系统测试的内容

1、验证（verification）

验证（verification）是保证系统正确的实现了一些特定功能的一系列活动，即保证系统做了我们所期望的事情。（Do the right thing）

1. 确定系统生存周期中的一个给定阶段的产品是否达到前阶段确立的需求的过程；
2. 程序正确性的形式证明。
3. 评审、审查、测试、检查、审计等各类活动，或对某些项处理、服务或文件等是否和规定的需求相一致进行判断和提出报告。

2、确认（validation）

确认（validation）是一系列活动和过程，目的是想证实在一个给定的外部环境中系统的逻辑正确性。即保证系统以正确的方式来做了这个事情。（Do it right）

1. 静态确认，不在计算机上实际执行程序，通过人工或程序分析来证明系统的正确性。
2. 动态确认，通过执行程序做分析，测试程序的动态行为，以证实系统是否存在问题。

系统测试的对象不仅仅是程序测试，系统测试应该包括整个系统开发期间各个阶段所产生的文档，如需求规格说明、概要设计文档、详细设计文档，当然系统测试的主要对象还是源程序。

5.2系统维护

系统运行后可能会暴露出一些错误、缺陷或者用户由于业务需要对系统有一些新的功能需求，这就要求系统在后续的使用中不断的得到完善和改进。因此，系统维护的主要任务就是保证系统的正常运转使系统的资源得到有效运用，并使系统的功能在后续的运行中不断的得到完善和扩充，以提高系统的工作效率和延长系统的生命周期。对系统的维护工作贯穿于系统的整个生命周期，维护工作的质量将直接影响到系统的使用效果和使用寿命。所有，系统维护是系统生存的重要保证。

5.2.1维护的内容

由于系统环境的变化或系统运行中发现了错误或问题以及随着用户的增多系统出现了性能问题，以及用户要求增加系统的某些功能等，提出了修改申请。系统维护工作包括以下三个方面：改正性维护、适应性维护、完善性维护。

改正性维护是在软件运行中发生异常或故障时进行。这些故障是由于开发过程中某些环节上的未被发现的错误造成的，在开发的末期所进行的测试也未能将其发现和修复。这些错误在某些特定情况下才能暴露出来。对已经发现的问题进行修改时，一定要十分谨慎。修改时一定要加上注释，如注明修改的日期、修改的原内容与修改后的内容以及修改人等信息。

适应性维护是要使运行的软件能适应外部环境的变动。由于计算机技术近年来发展得越来越快，摩尔定律越来越不适用，几乎6-12个月就出现一代新的系统和硬件，建立在硬件和操作系统之上的应用系统，其使用年限远远超过6-12个月，这要求应用软件能跟上发展的趋势而作出相应的修改。

完善性维护是为扩充软件的功能，提高软件的性能而进行的开发的维护工作。在系统使用一段时间之后，用户提出了新的要求，要求在现有软件基础上进行扩充开发新的功能；以及随着用户的增加，系统的并发量越来越高，需要对系统进行性能上的优化等所需的开发工作。

当前，系统维护工作还存在一定的困难，由于整个系统开发过程是由一个人完成的，难免会存在个人能力上的技术瓶颈，对后期维护经验不足，在人力物力上存在一些实际困难。这样一来，导致维护力量薄弱，常常由于考虑不周，在维护的过程中可能带来新的问题或引入新的差错。

5.2.1系统维护的管理

系统维护工作不仅是技术性的工作，还需要大量管理工作与之配合，才能保证维护工作的质量，一般维护过程如下：

1. 维护人员提出维护修改建议或要求。
2. 管理部门进行审查和分析，并对由于修改所带来的风险作充分的估计。
3. 根据审批的修改方案进行修改并作严格的测试。
4. 更新文档资料的相关部分。

结 论

经过几个月以来紧张有序的毕业设计，使我对管理信息系统的开发步骤、方法有了深一步的认识，对开发流程有了切身的体会，巩固了信息管理系统所学的各项知识，同时也坚定了继续投身于信息系统学习、研究与开发的信心和决心。并且有机会对本专业的基本理论、专业知识和基本技术有了更深入的了解、体会和运用，使得我不仅仅对专业理论的理解达到了一个新的高度，最主要的使我把四年来所学到的知识得到了系统性的升华，达到了真正的学以致用，不但提高了处理实际问题的动手能力，而且学到了很多课堂以外的东西。随着毕业设计接近尾声，我设计和开发的海豚湾美甲店美业管理系统管理信息系统也已经初具规模，尽管系统还有一些不完善之处，但这是一次有价值的尝试，每一个界面的设计，每一个功能模块的规划，每一行代码的编写都付出了我的全部心思。

1.系统的特点

* 本系统为全中文界面，功能全，易操作，交互体验好。本系统的界面风格Bootstrap风格。
* 有相应权限的用户可自由地设置系统中的各类相关参数和各种项目代码。
* 系统中具有强大的操作员权限、密码管理等功能模块，确保系统运行安全。
* 有效的查询和报表功能，全面、及时地反映当前状况。
* 系统结构严谨、性能稳定、使用方便。
* 速度快。
* 投资少。
* 学习容易。不具计算机经验的管理人员和操作员只要具有计算机管理的基本知识即可在较短的时间内掌握；系统采用图形界面和向导式操作，本着“功能越复杂，操作越简单”的原则设计，易学易用。
* 维护简单，系统运行后几乎不需要专业系统管理员维护，减少开支。

使用本系统可以提高设备可靠性，减少每天需要涉及到的相关工作量，减少工作人员的劳动强度，减少办公耗材，尽可能的提高工作效率，可以减少人力物力的浪费，提高的现代化管理水平。

2.系统评价

系统评价主要是指系统建成后，经一段时间的运行后，要对系统目标与功能的实现情况进行检查，并与系统开发中设立的系统预期目标进行对比，及时写出系统评价报告。

系统维护与评价阶段是系统生命周期中的最后一个阶段，也是时间最长的一个重要阶段，就像汽车的维护工作好可以延长汽车的使用寿命和提高其使用效率一样，系统维护工作的好坏可以决定系统的生命周期的长短和使用效果。

系统评价：本系统的投入运行基本满足了实验室及设备的管理、查询及预定。运行状态平稳，其安全性、稳定性、可维护性及功能配置都表现良好。表明系统的各项指标达到了设计的要求。

目前，本系统由于硬件条件的限制还仅作为一个独立运行的系统，并未真正的实现联网工作状态，数据转换过程中还需要人工操作，另外随着需求的变化，功能上还要尽可能完备，有待继续扩展和完善。