VCampus虚拟校园项目报告

组长：潘东元 09014231

邮箱：641234230@qq.com

组员：王铎 09014222

丁子昊 09014229

许旖 09014139

乔梁 09014124

曹心成 09014116

*2016/9/15*

***一、前言***

*1.1 项目背景*

作为短学期的一项软件实践项目，虚拟校园系统首先更贴近我们校园学生的生活学习，通过日常的使用本校的校园系统，我们可以更加方便的给出我们对于软件的需求以及其可行性分析，也更加容易给出我们对于软件的设想。同样结合我们对于java语言的理解，帮助我们巩固我们对于该语言的现实应用。通过实践，增强我们的软件工程意识，和在团队中的协作方法。

*1.2 项目设想与准备*

我们设想的虚拟校园系统是一款针对教师、学生、管理员的一套在线校园系统，在现在校园信息化的大背景下，作为学生老师群体都需要一款方便、快捷、有效、健壮的校园系统去统一管理教务、图书馆、基本信息等，其信息化、网络化与数据库结合的形式可以降低人工产生的效率低下以及潜在的问题，同样软件的便捷有效及其友好简介的界面和人性化的逻辑可以简化用户的维护，查询，修改信息的时间代价，是现代化校园信息化最核心的一部分。

项目的准备：学习java界面开发，数据库以及网络通信的相关知识，Java编程规范。

*1.3 人员具体分工*

组长（09014231）：

潘东元：数据库模块，模块的整合调试、项目的架构，登陆模块的设计，界面的部分设计与美化。

组员：

王铎（09014222）：系统的总体架构，项目的需求分析，基础类的设计以及各个模块的整合、调试，所有模块的连接工作。

曹心成（09014116）：GUI的主页、图书馆、基础信息以及商店模块的开发，虚拟校园系统使用报告，项目总结，设计说明书的撰写。

丁子昊（09014229）：客户端GUI的总体架构，GUI课程、登陆模块的开发，界面的美化，项目的整合调试。

许旖（09014139）：管理员系统的开发（包括GUI和数据库操作函数），网络连接模块。

乔梁（09014124）：GUI生活服务模块的开发。

*1.4 功能简介*

虚拟校园系统提供给了学生以及系统管理员一个方便快捷的校园运营维护系统。针对学生日常的学习生活需求。

在主页模块提供了教务，社团，体育等相关信息的快速浏览，方便学生群体第一时刻掌握校内学习生活娱乐的最新推送，并且信息以图片形式的呈现，相比于枯燥的文字更加吸引用户的关注。

在选课系统中给出了四个子模块，分别是课程表模块，直接直观地将选信息以图表形式呈现出来。在成绩查询模块，根据学期的不同查询所选课程，提供了该学期各个课程所获成绩，总学分以及基点。选课系统分为选课和查询已选两个模块。

在图书馆模块，分为三个子模块，分别是借阅历史查询，方便用户了解当前的借阅书目，及时归还书籍，；在书籍查询模块，分为按作者查询和按书名查询；第三个子模块是借阅历史，提供给用户及时查看自己曾经借阅的书籍。

在生活服务模块，提供给了用户基本信息的查询与维护功能，以及一卡通服务的相关实现，有为一卡通进行充值，挂失以及解冻等操作。在宿舍模块，提供给了用户查询宿舍分数以及故障报修的服务。同时提供了医疗系统，可以提供给用户查询病例功能，并且可以预约诊室。

在个人商店模块，虚拟校园系统提供给了用户大量的商品选购，所有的商品信息都存储在服务器的数据库中，可以定时的更新。购买商品时可以将商品放入购物车中，等选购完成之后一起进行结算。在第三个子模块中提供了交易记录的查询。并且购物车以及商品选购的界面是一个动态管理的界面，可以随时根据后台数据库的更新信息进行刷新，并且刷新相应商品的大图。

***二、项目需求***

*2.1 整体项目分析（需求）*

虚拟校园系统的主体用户是学生，管理员则有直接修改数据库的权限。因此通过直接分析用户的需求，可以得到我们项目整体的需求分析。

首先是针对主体用户学生：（表一）

|  |  |  |
| --- | --- | --- |
| ***学生需求*** | ***子模块分析*** | ***具体实现*** |
| 选课系统 | 提供选课以及退课功能，选课完成后以课表的形式呈现，选完课程之后就无法再次选择该门课程 | 将该模块归并于选课以及成绩查询系统，设在同一个标签下，方便对照先前学期的成绩 |
| 成绩管理系统 | 提供成绩查询，基点查询。根据所选的相应学期，列出课程信息。课程学分以及所获得的成绩，并且根据这些信息在后台计算出该学生在所选学期的基点以及总学分情况。 | 成绩查询系统对于学生用户的使用特别重要，所以此功能作为主页后的第二个标签与选课系统合并，成为学生用户点开就可以立即查询的功能模块。 |
| 图书馆系统 | 图书馆主要的功能是显示该学生的借阅信息，系统自定义学生借阅书籍的上限是八本。并且提供续借服务，可以使归还时间增加一个月。查询功能提供了按作者查询和按书名查询两种，支持模糊查询。最后的功能提供给了学生查询自己所有借阅信息的功能。 | 图书馆拥有三个子模块，自己占据一个标签，标签首页列出了学生所有的借阅信息，方便查看。 |
| 一卡通系统 | 一卡通系统提供了为一卡通充值功能，并且考虑到现实一卡通的的种种异常操作，提供了挂失解冻等功能。 | 一卡通模块并入生活模块。 |
| 宿舍管理系统 | 宿舍系统管理系统为用户查询宿舍信息提供了功能，并且允许用户进行保修，和查询宿舍卫生成绩。 | 宿舍管理模块并入生活模块。 |
| 基本信息系统 | 基本信息模块为用户提供了查询和维护个人基本信息的功能。 | 基本信息模块并入生活模块。 |
| 网上商店系统 | 网上商店系统为用户提供了可以时时刷新的商品列表，随时查看商品的大图，并且可以加入购物车，在购物车中，用户需要对商品的数量进行更改，并且进行付款操作。最后的交易记录讲显示用户所有的交易记录信息。 | 网上商店按照需求分为三个子模块，可以归并为一个标签之下。用户点击网上商店后，应该直接显示商品列表，选购商品完成后可以点击进入购物车进行所有商品的结算。 |

其次是针对最高权限者管理员的需求：（表二）

|  |  |  |
| --- | --- | --- |
| ***管理员需求*** | ***子模块分析*** | ***具体实现*** |
| 管理学生信息 | 提供界面完整输入学生所有信息，并进行删除操作，展示所有学生信息 | 提供完整的输入框 |
| 课程模块 | 提供界面完整输入所有课程信息，并进行删除操作，展示所有学生信息 | 提供完整的输入框 |
| 商品模块 | 提供界面完整输入所有商品信息，并进行删除操作，展示所有学生信息 | 提供完整的输入框 |
| 图书馆模块 | 提供界面完整输入所有图书信息，并进行删除操作，展示所有学生信息 | 提供完整的输入框 |

*2.2 项目模块分析*

2.2.1 数据库模块

数据库模块需要提供函数，让服务端调用写入数据或者查找数据。通过Access软件的帮助，建立合理的数据表，减少数据层耦合的同时提高数据读写效率。

2.2.2 网络通信模块

网络通信模块需要建立客户端与服务端统一的通信标准客户端通过网络发送并接收服务端的消息，服务端需要可同时处理多个客户端的连接请求，并通过数据库操作完成响应功能。

2.2.3 界面模块

界面模块需要建立每个界面设计的统一规范，建立易于管理的模块跳转设计，利用逻辑关系的跳转连接各个界面，同时读取或者写入客户端的数据。在基本功能的基础上，美化界面，提高用户友好度。

2.2.4 基础类模块

基础类模块需要建立合理的基础工具类，与通信标准保持一致，便于客户端与服务端的通信，同时基础类对应与每个界面的功能与数据库数据表的设计，减少数据的耦合。

*2.3 项目可行性分析*

根据上述的项目需求，我们可以得出所有模块所有操作的各种不同功能，对应于不同功能的实现。我们将基于模块功能的不同，从一个功能所执行的周期，经历的不同工种所完成的工作来分析该模块的可行性研究。

对应于整体项目，通过上述需求分析（见表一、表二）所获得的模块分类，我们将他们相对独立了开来，模块之间联系相对对立，可以进行独立的开发（除了商店的消费系统与一卡通的充值系统之间存在联系之外）。从纵向进行比较，因为上述横向模块之间相对较为对立，纵向从数据流向来看（如从GUI发送请求，有网络通信负责打包传输，再有服务端接收并且进行数据库操作，再有反向数据流传回客户端，实现GUI的显示），软件的开发可将小组成员从纵向划分，完成分时段完成具体横向的各种现实功能的实现，再有人员负责纵向整合，这样特定纵向模块的开发人员可以负责相对独立的模块操作，并且通过不断深入，更加完善如GUI此特定模块设计与响应操作的实现。

通过总体系统的可行性分析，我们大致明确了项目项目实现的整体思路与具体操作目标，实施方案就转为相应模块的负责人深入学习与实践相应模块的基础知识，做出相对独立的模块，供专门人员进行整合，使得系统的可拓展性，模块对立性以及系统的针对相应模块可能的容错率都有显著提高。

因此从整体系统考虑我们的设计思路、任务分配 是可行的，接下来需要我们针对不同模块进行细节方面分析可行性。

*2.4 整体项目分析*

2.4.1数据模块

数据模块通过利用Access建立数据表，通过关系型数据库存储数据，通过java所包含的数据库驱动进行数据库的连接，利用编写SQL语言代码实现功能。

2.4.2 网络通信模块

网络通信模块利用java提供的Socket类与ServerSocket类实现客户端与服务端的数据交流，建立Message类作为通信统一标准，包含所传递的数据与标识码，编写Client类与Server类作为实现通信的功能接口类。

2.4.3 界面模块

界面模块使用java的Swing包与Only-feel-0.3包实现界面类的设计与界面逻辑之间的跳转，同时使用控件的事件响应类完成页面跳转。

2.4.4 基础类模块

基础类模块为各个对象设计所包含数据的类，对应于学生信息，图书馆信息，课程信息，宿舍信息，一卡通信息，医疗信息等，加载于Message类中。

***三、项目设计与开发***

*3.1 项目开发环境*

java集成开发环境：Eclipse Mars，Eclipse Kepler，Eclipse Neon

软件开发工具包：jdk1.7

数据库：Microsoft Access2013，Access2010

GUI插件：Windows Builder

*3.2 项目总体架构*

3.2.1 系统逻辑

3.2.2 通用层

|  |  |  |
| --- | --- | --- |
| 类名 | 说明 | 所在文件 |
| AccountInfo | 账号信息类,提供登陆信息 | AccountInfo.java |
| ApartmentInfo | 提供用户的所在宿舍信息 | ApartmentInfo.java |
| BasicInfo | 基础信息类，提供用户的个人基本信息 | BasicInfo.java |
| Book | 书类，使所有其他书类的成员 | Book.java |
| BorrowBook | 用户所借图书的集合 | BorrowBook.java |
| Clinic | 预约诊室类，提供可以预约的诊室集合 | Clinic.java |
| Course | 课程类 | Course.java |
| EcardInfo | 一卡通信息类 | EcardInfo.java |
| EcardRecord | 一卡通记录类，提供一卡通的交易记录 | EcardRecord.java |
| Goods | 商品类，作为其他所有其他商品类的成员 | Goods.java |
| Grade | 学生成绩类，存储成绩信息 | Grade.java |
| MedicalHistory | 医疗记录类，用于查看用户就诊情况 | MedicalHistory.java |
| Messgae | 信息类，用于数据库、服务器、客户端之间的数据传递 | Messgae.java |
| Person | 用户类 | Person.java |
| Shopbasket | 购物车类，存储用户加入购物车的所有商品 | Shopbasket.java |
| ShopRecord | 购物记录类，用于记录用户所有的购买记录 | ShopRecord.java |

*3.3 项目设计思路*

3.3.1数据模块

基于各个基类，建立对应的数据库表，对应各个基类的数据成员建立表中的字段。并且某些关联到具体用户的数据表需要包含用户的识别码，这里我们使用一卡通代替。对数据库的操作我们建立了一套完整的操作码，客户端通过message发送数据中包含操作码，在服务端解析识别操作码后调用该操作码的操作函数，对数据库进行对应操作。在从数据库读取数据要想客户端发送时，直接对message中封装的类的对象等进行操作，由于JAVA对象传递默认是引用传递，于是在操作函数修改后，直接返回发送。

3.3.2 网络通信模块

1. 主要目的是实现客户端/管理员端和服务器端的通信，因此先建立双端用以通信的函数。
2. 又因为要实现多用户同时操作，所以考虑使用多线程技术，在服务器端为每个用户操作分配一个线程进行处理。
3. 而为了方便客户端和服务器端的使用，还需提供统一的网络接口，使其可以通过传送操作码即可进行相应操作。
4. 又双端处理的多为对象，因此在网络上进行对象的传输简单明了，所以这里还实现了序列化后对象的传送

3.3.3 界面模块

1. 界面的美化操作，合理布置界面，给用户简洁友好的界面布局以及人性化的跳转响应逻辑。

2. 对界面的设计相对独立的分开各个模块，不同模块对应一个子包，并且拥有一个或多个类，让不同界面之间相互影响和冲突降低到最小。

***四、项目关键技术***

数据库连接使用了JDBC-ODBC桥。数据库连接要求使用JDK1.7一下的版本，因为1.8已经舍弃了JDBC-ODBC桥连接的方式。另外由于驱动问题，应当使用与系统统一位数的JDK，ACCESS。

操作码与数据库操作：这部分使用到的SQL语句并不是很多，但是设计的越完善，这部分代码就会越多。根据操作码，进行操作码要求的功能，对数据库的多张表进行操作并返回数据以及对应的操作码（操作执行的状态）。

在网络通信部分，对于上文提到的设计思路1，采用java socket和serversocket即可实现。

对于设计思路2，重载一个thread的类成serverthread，每到来一个请求，就在在server端调用serverthread进行相应操作。

对于设计思路3，在serverthread中对不同的操作码进行识别，调用不同的数据库函数进行操作，并返回用户不同的回应。

对于设计思路4，将所有要进行传送的对象序列化，调用ObjectOutputStream和ObjectInputStream建立输出输入流。

GUI主界面及框架

1．对JButton重载，实现normal,focus,pressed三种状态下按钮的不同图标，实现动态效果。

2．采用CardLayout，通过一级Tab，二级Tab的点击实现不同功能的容器间的跳转。

GUI课程模块：

1. 利用Graphics给课程表画课表线
2. 继承TableCellRenderer以及DefaultCellEditor，往JTable中加入Button渲染器，并实现相关响应函数。
3. 利用DefaultTableModel对表格进行管理，动态增删表格的内容。

GUI主页模块：

利用JLayeredPane对控件的层次进行处理，从而做出动态的效果

GUI购物车模块：

利用JTable的渲染器实现Table中加入JPanel，从而达到动态增删购物车内容的效果。

GUI服务端界面

多线程防止服务端启动后停止接受鼠标响应，调用InetAddress来获取本机IP地址

***五、项目经验及存在问题***

*5.1 经验*

谈及经验，不得不说我们小组的任务的分配以及时间进度的安排。我们小组在前期的时候就有对整个项目有了一个非常清晰的架构和合理的分工。首先是软件整体的架构，基础类和网络通信优先设计，提供给了数据库以及界面学习并且有媒介可以测试的环境，同样我们从一开始就有明确的分工，明确的要求会精确到某一个人的某一天需要完成的工作。因此给我最深的影响就是我们小组一直以来就是以高速与高效在运转，并不像有些小组会在中期之后突然开始临时抱佛脚，因此也会给我们有更多的时间去调整。我相信所有这一切的优势都依赖于从最开就开始着手规划项目的执行，并且早早就得出了对于软件设计的框架结构，因此任务可以及时的分配到个人头上。

*5.2 存在的问题*

我们小组的沟通主要集中在课堂上，课后并不是所有人都可以聚集在一起完成代码的任务，往往课后的缺乏交流会导致一些无用功的产生。因为在组长分配完成任务之后，可能组长期望的的实现并不一定是组员所能达到的实现，除了实现能力本事之外的因素之外，通过交流可以互相解决好多问题，没有必要为了一些小的问题而花很久的时间；除此之外就是没有达到组长期望的样子之后返工所花费的时间。就拿GUI部分为例，首先是我们的界面风格一开时没有统一起来，导致后面为了统一风格重做界面的事情常有发生，其次就是界面的逻辑跳转方面也是需要事先商量好的，因为在发送操作请求给服务器端的时候，需要统一操作的编码格式，以及数据类型。

*5.3 团队沟通和工作方式*

团队的沟通方式主要集中在课堂的讨论，并且会大家集中在一起写代码或者调试程序。主要集中在宿舍副楼完成开发工作。

工作方式以模块分工完成项目任务，有专人负责代码的整合工作。

*5.4 技术理解*

数据库方面：主要是对功能的实现，要怎样建立数据库才简单而又不冗杂，以及为了实现某个功能，对数据库操作的逻辑问题，怎样有序的操作。

网络通信：对于基本网络通信的建立，方法上采用最平实的socket和serversocket对象进行处理。简单，迅速，好用。而对于多线程的处理，建立服务器端的线程类，并在操作时调用该类进行处理即可。对操作码的识别，可简单通过switch语句处理。对象的传递则通过序列化，和封装好的输入输出流进行。

GUI设计：首选要搭出界面一开始的布局，不能边写代码边设计布局，事先设计好之后，可以考虑布局的合理性以及相关响应如何实现，单纯的只是在框架上堆砌控件并不会考虑到用户的用户体验和逻辑，会导致重复工作。其次就是界面是一个没有上限也没有下限的工作，理解的越深入，可以做到更加漂亮的界面，这需要不断学习。

*5.5 工作方式的解决*

工作方式在可行性分析中也分析过在我们分析完成项目的整体之后，所确定的每个人负责横向模块的思路，有专门人员负责GUI，有专人负责数据库，有专人负责网络通信等等，这样子的工作方式可以让我们专精于一项模块的开发，可以更加深入的了解该模块的开发过程，更加可以使一样模块做得更好，通过以前的经验。后期GUI方面压力较大，完成其他模块的人员一同参与最后GUI的开发与美化工作。