|  |
| --- |
|  |
| 心窗密语 |
| 移动互联网环境下基于大数据的情感分析系统 |

**目 录**

[一、概述](#_Toc398910541) 1

[1.1 选题背景](#_Toc398910542) 1

[1.2 项目意义](#_Toc398910543) 1

[二、可行性分析和目标群体](#_Toc398910544) 2

[2.1 可行性分析](#_Toc398910545) 2

[2.2 目标群体](#_Toc398910546) 2

[三、项目功能和原型设计](#_Toc398910547) 2

[3.1 总体功能结构](#_Toc398910548) 2

[3.2 具体功能模块设计](#_Toc398910549) 3

[3.3 界面设计](#_Toc398910554) 7

[四、项目实现、特色和难点](#_Toc398910557) 11

[4.1 项目实现](#_Toc398910558) 11

[4.2 特色分析](#_Toc398910559) 12

[4.3 难点和解决方案](#_Toc398910560) 13

[五、团队介绍和人员分工](#_Toc398910561) 13

[5.1 团队介绍](#_Toc398910562) 13

[5.2 人员分工](#_Toc398910563) 13

[六、其他](#_Toc398910564) 14

[6.1 开发结束后的感触](#_Toc398910565) 14

[6.2 不足之处和今后设想](#_Toc398910566) 14

[6.3 用到的第三方代码或组件](#_Toc398910567) 15

[七、致谢](#_Toc398910568) 15

[八、需求规格说明书](#_Toc398910568) 16

[九、概要设计说明书](#_Toc398910568) 27

[十、详细设计说明书](#_Toc398910568) 41

[十一、测试说明书](#_Toc398910568) 49

1. 概述

1.1选题背景

近年来，随着社会经济的飞速发展，在人们的生活节奏加快的同时，心理问题也在日益增加着对人们的影响。从疾病发展史的角度来看，人类已经从“传染疾病时代”、“躯体疾病时代”进入了“心理疾病时代”。据相关调查显示，我国目前有1600万精神疾病患者，而青年大学生是主要群体之一。在大学生群体中，16.0%-25.4%的大学生存在不同程度的心理问题，其中约有2.8%存在严重的心理疾病。在本科生中因心理疾病休学的人数占因病休学总人数的37.9%，因心理疾病退学的人数占因病退学总人数的64.4%，均居首位。由严重的心理问题或心理疾病引发的出走、自杀、凶杀等恶性案件在各高校中更是时有发生，各高校也被推到风口浪尖。因此，如何有效加强大学生的心理健康教育以及对大学生的心理状态进行预测和控制，成为了高校需要解决的最重要和紧迫的问题之一。

1.2项目意义

为社会培养和输送能够适应未来世界的建设者是高校的首要任务。这些全面性高级人才必须具有坚忍不拔的意志、艰苦奋斗的精神、积极乐观、奋发进取等良好品质。全面提高大学生的心理素质既是社会发展的需要，也是未来社会对人才素质的要求之一。

在如今青年大学生心理健康问题层出不穷的情况下，我们的应用——心窗能够有效地对大学生心理状态进行预测和控制，避免由于严重心理问题引发的恶性案件的发生，让大学生在良好的心理状况下茁壮成长，成为社会需要的人才。本应用脱离了传统式的心理咨询模式，将其与互联网社交平台紧密结合，通过关注学生的微博，分析出学生近期的心理状态变化，找出那些可能存在潜在心理问题的同学，帮助高校的心理医师或教育界工作者和研究人员了解学生们的心理状态、及时地关注到心理状态异动的同学。  
 同时，随着云计算时代的来临，大数据不断吸引着各界人士的关注。大数据充分体现了合理应用数据的重要性。大数据的关键并不在于数据量的庞大，更重要的是对这些数据的处理。大数据相关工具分几个方面，存储方面比如MySQL和Oracle，分析方面比如Hadoop和Oracle Exadata，这些工具之间既有共同点，也有些许不同之处。本应用基于大数据的挖掘和分析，通过采集大学生的社交平台（微博）所发信息，设计科学的算法，为大学生建立其性格模型，并根据其最近几天所发信息，分析其近期心理状况的变化。

二、可行性分析和目标群体

2.1 可行性分析

技术可行性：目前，我们团队掌握的技术可以完成本应用的开发。其中数据的获取方面可通过爬虫技术或通过官方接口实现。如今，爬虫技术已相对成熟，并有相关文献供参考及学习使用，通过学习爬虫技术可以实现对用户微博数据的获取。一些社交平台，譬如新浪微博等，他们向外界用户提供接口来实现用户消息数据的获取，供个人开发者使用。我们也可以通过调用官方接口来实现数据的获取。数据挖掘和分析方面将采用大数据分析方法。本组成员对数据库的使用都有较深的理解，也修习过数据挖掘课程。在此基础上，通过查考文献资料，设计出科学高效的算法，建立用户性格模型及分析用户近期心理状况变化是可实现的。另外本应用采用C/S模式的搭建方案，基于C/S架构的软件，其开发模式和方法均已十分成熟，有许多经验可以借鉴，且本小组成员均有丰富的C/S架构软件的开发经历。

市场可行性:就像前文提到的，如今我国大学生心理健康的状况不容乐观，各高校急需全面提高大学生心理素质以及避免由严重心理问题引起的恶性案件的发生。而本款应用可以帮助高校有效地对大学生的心理状况进行监测控制，从而让心理医师或教育界工作者及研究人员及时地对存在心理问题的同学进行疏导，避免心理问题加剧，酿成极端事件。可以说有大学生的地方，对我们软件的需求就会存在。据国家统计局发布的数据，2014年研究生在校生达到184.7万人，普通本专科在校生达到2547.7万人。因此我们的应用的市场需求是相当大的。

**2.2 目标群体**

本项目的目标用户群体是高校学生、教育界工作者、研究人员及心理医师。用于对大学生的心理状况进行监测控制。

三、项目功能和原型设计

3.1 总体功能结构

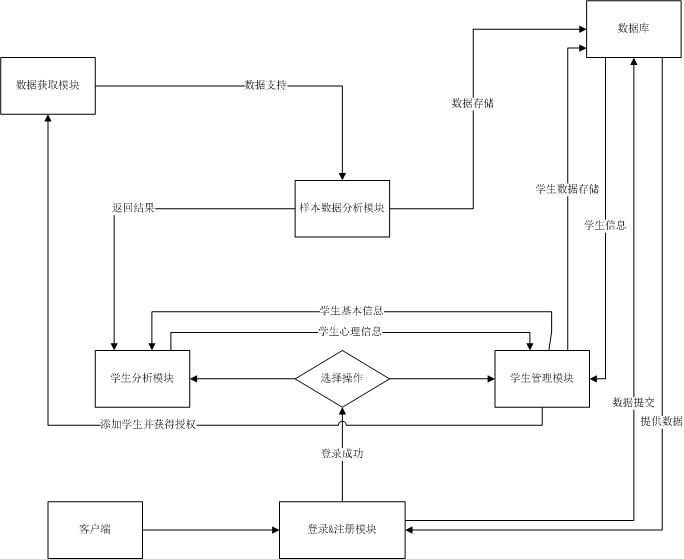


图3.1 总体功能结构图

本项目的总体功能结构图如图3.1所示。

用户首先进行账号注册，填写个人信息完成后提交即注册成功。进入主界面点击添加学生并请求学生授权后，服务器接收到授权信息并通过官方提供的接口获取到用户近期的数据，然后对这些数据进行挖掘与分析得出学生的心理状态，将结果返回给客户端。用户登录后可以管理学生并查看学生的心理状态。

3.2 具体功能模块设计

（1）登录注册模块

注册登录模块主要分为两部分:注册、登录。

注册模块中，用户输入注册相关的个人信息，比如用户名、密码等，作为与其他用户的区分。输入完成后所有信息将会被发送至服务器并储存在数据库中。

登录模块中，用户需要使用已注册并进行授权的账号和密码进行登录，客户端将登录请求和输入信息发送至服务器，服务器接收到请求后将输入信息与数据库中的注册信息进行匹配，匹配通过则将登录成功的提示信息反馈给客户端。

图3.2.1为注册&登录模块的功能结构图：

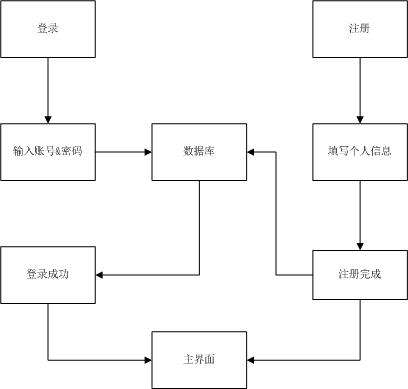


图3.2.1 注册登录模块

（2）数据获取模块

用户添加学生后，会对该学生发出授权请求。请求通过后，学生授权信息会被发送至服务器，服务器调用官方接口进行学生数据的获取。然后将获取到的数据保存并对该数据进行挖掘和分析。

图3.2.2为数据获取模块的功能结构图：

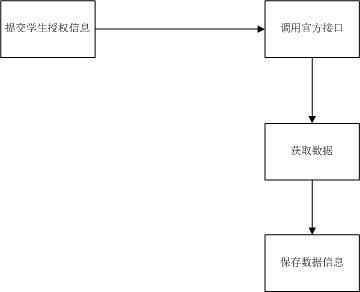


图3.2.2 数据获取模块

（3）样本数据分析模块

在样本数据分析模块中，数据获取模块中获取到的用户数据将会被用到。首先对学生的微博等社交平台的动态数据进行关键词的拆分与提取，将每一句话按照词性进行拆分，然后将情绪类形容词、程度副词等需要进行进一步分析的关键词提取出来，结合心理学相关知识进行分析。我们将人类的情绪以及积极或消极的程度用数字来表示，每一个词都有相应的数值，通过这些数值之间的运算，我们可以得出用户心理状态的数值表示，这就是关键词的分析过程。

图3.2.3为样本数据分析模块的功能结构图：

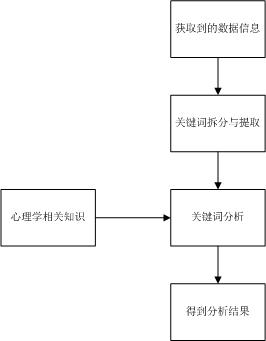


图3.2.3 样本数据分析模块

（4）学生分析模块

在学生分析模块中，首先会接收到样本数据分析模块的分析结果信息。我们通过对学生社交平台的动态信息进行分析，得出一个心理状态的健康值。学生分析模块根据参考值分析每一位学生近期的心理状态并将结果存储在数据库中，并且将当前用户所添加的所有学生的健康值进行归类统计，最终将结果以图表的形式呈现出来。用户可以根据所呈现的结果看出学生的心理状态总体趋势。

图3.2.4为学生分析模块的功能结构图：

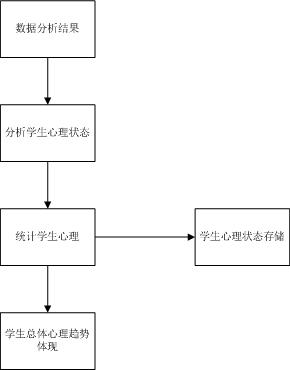


图3.2.4 学生分析模块

（5）学生管理模块

在学生管理模块中，用户可以查看每一名学生的心理状态详情，还可以直接查看其微博等社交平台近期发送的消息；同时，用户也可以添加学生或删除学生，添加或删除的同时数据库中的内容也会相应的随之改变。用户在添加学生时，我们会给该学生的社交平台账号发送授权请求信息，通过后即可进行学生相关信息的获取。

图3.2.5为学生管理模块的功能结构图：

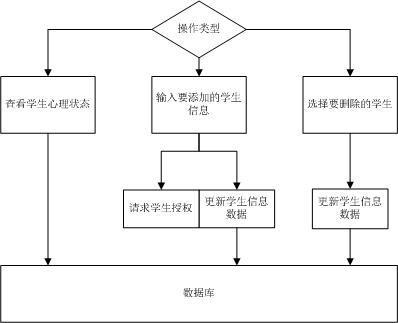


图3.2.5 学生管理模块

3.3 界面设计

本应用为一款数据分析类应用，主要用于对学生微博所发布的消息进行分析，分析完成后将结果整合成图表的形式供教育工作者进行查看。图表分为两种，一种是学生情绪分布图，另一种是学生情绪走势图。

其中学生情绪分布图中展现了消极、正常、积极三种情绪分别对应的微博消息数占总消息数的比例；学生情绪走势图中，横坐标代表所有学生的微博消息序号，纵坐标代表学生的情绪指数，正值代表积极情绪，负值代表消极情绪，0代表正常情绪，该图展现了近期学生情绪的走势。

图3.3.1和图3.3.2为“全校统计”页面中的全校学生分析结果统计图。

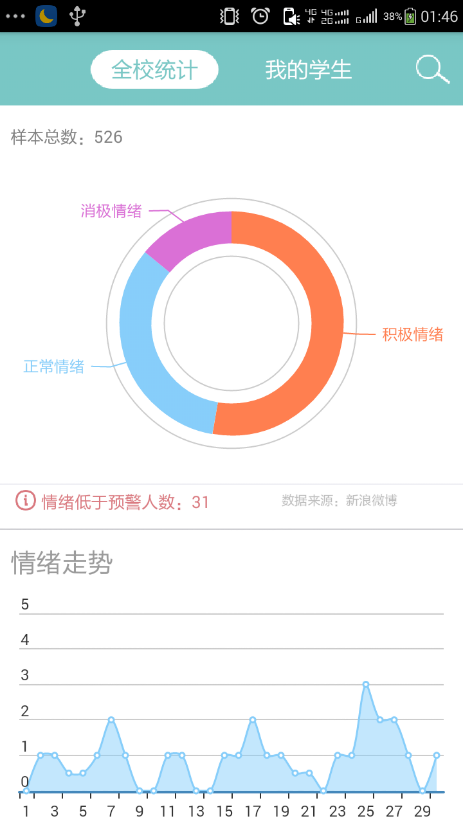
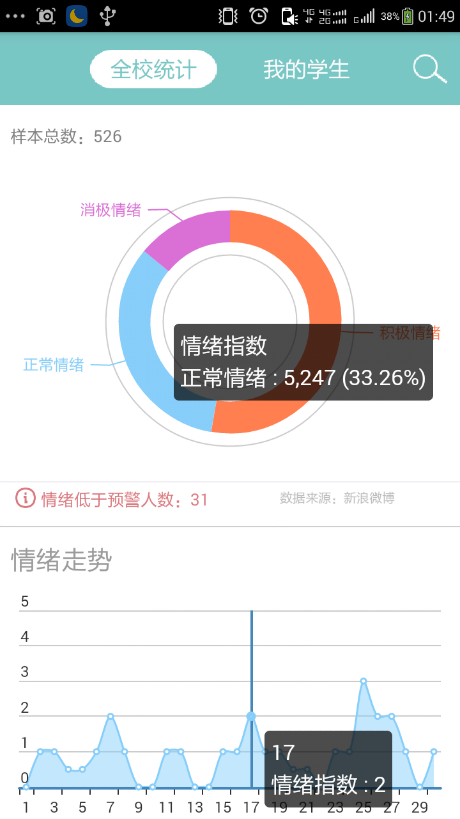
 

图3.3.1 “全校统计”页面 图3.3.2 图表详情

图3.3.3为“我的学生”页面中当前教师用户已添加的学生分析结果统计图。

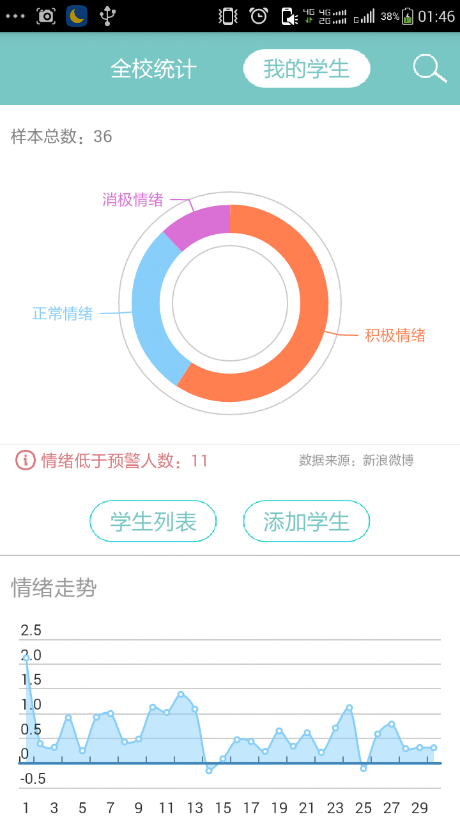


图3.3.3 “我的学生”页面

本应用的亮点在于我们可以对分析结果进行多种形式的查看，可以通过动态拖拽等形式使数据形成更加鲜明的对比。如图3.3.4所示。

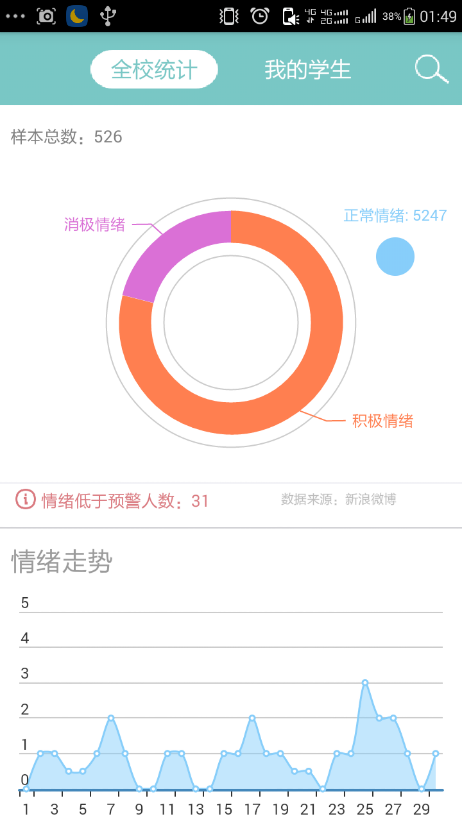


图3.3.4 动态操作数据

教师用户可以在学生列表中查看当前已添加的学生，并且可以录入新的学生，如图3.3.5和图3.3.6所示。

图3.3.5 学生列表 图3.3.6 添加学生

点击“我的学生”中的预警信息，可以查看所有近期情绪较为低落的学生。点击学生可以查看近期该学生情绪的具体状态，进入后点击右上角即可查看微博详情，如图3.3.7、图3.3.8和图3.3.9所示。

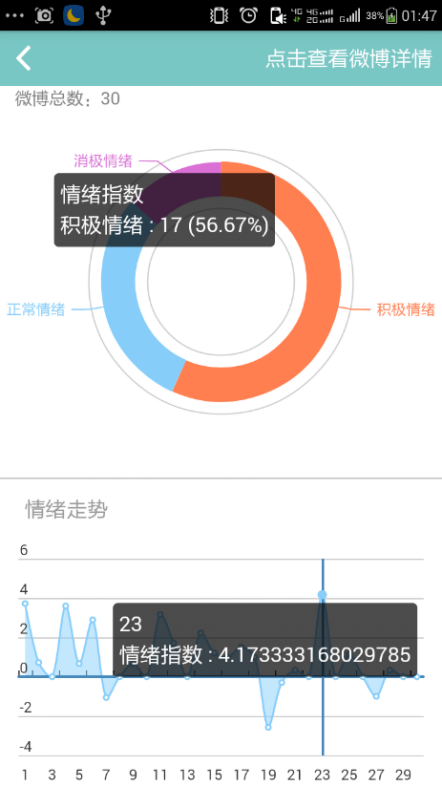
 

图3.3.7 情绪低落学生列表 图3.3.8 情绪低落学生具体状态



图3.3.9 微博详细内容

在“全校统计”或“我的学生”页面点击右上角的搜索按钮即可通过微博账号对关注的学生或感兴趣的微博用户进行查询，同时还可以通过“热门搜索”直接对热门用户进行查询。查询完成后，用户可以查看被查询用户近期的情绪状态，以及微博详情。如图3.3.10、图3.3.11和图3.3.12所示。

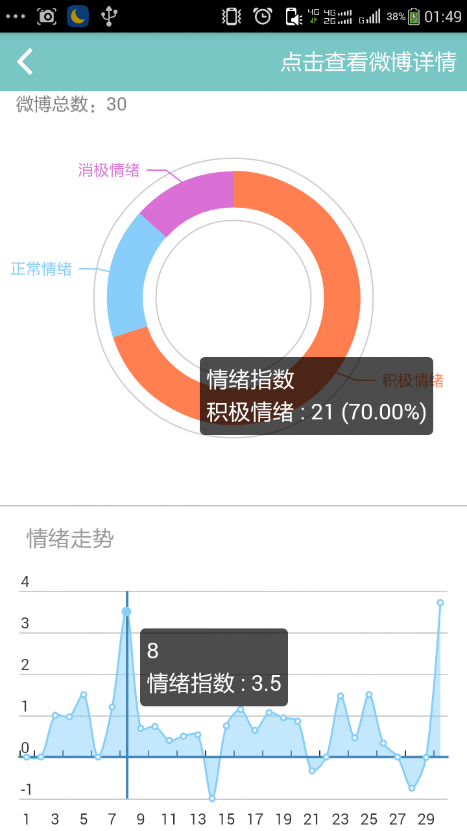
 

图3.3.10 搜索页面 图3.3.11 目标用户近期情绪状态



图3.3.12 目标用户微博详细内容

四、项目实现、特色和难点

4.1 项目实现

项目的移动终端版本的整体实现分为三个部分：文本预处理、文本倾向性分析以及情感输出显示。

文本预处理：首先通过学生提供的微博账号对相应的微博消息进行爬取，爬取成功后对微博数据进行分词和词性进行标注并进行去噪声处理，然后使用中科院情感分词系统对处理后的句子进行切分。

文本倾向性分析：对文本预处理得到的词汇进行进一步分析。一方面，我们使用切分后的句子进行句子分析；另一方面，我们将切分后的句子中的情感单元提取出来，使用微博文本敏感词词典以及HowNet情感词词典对情感单元进行短语倾向性分析。对两方面分析得到的数据进行综合分析，最终得到句子倾向性分析结果。

情感输出显示：将文本倾向性分析中得到的句子倾向性分析结果输出并发送至客户端，客户端对数据进行处理后展示出来。

作品的整体实现思路如图4.1所示。

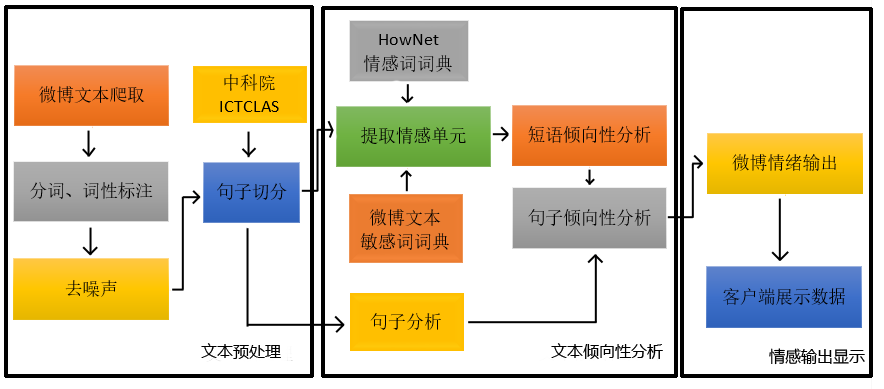


图4.1 作品实现思路示意图

4.2 特色分析

社交平台（微博）是大学生发表感情、记录心情的地方。然而，这些发布在社交平台上的信息并未引起太多人的注意。

本应用基于大数据的挖掘和分析，使这些大量的复杂的数据有了意义。通过这些数据，我们可以对学生的心理状况进行监测，一旦心理医师或教育界工作者或研究人员发现某些学生的情绪异动，可以持续的关注该学生或者对其进行心理疏导工作。同时，了解同学平时在社交平台上发布的信息，为其建立性格模型也有助于后续的疏导学生的工作。

另外，本应用脱离了传统的心理咨询的模式，不是等着学生来进行心理咨询，而是主动出击，持续监测学生的心理状况，将严重的心理问题扼杀在摇篮中。它相对于传统的心理质询服务，具有预见性、持续性、实时性、便捷性等特点，也可避免学生对传统心理咨询的抵触心理。

4.3 难点和解决方案

（1）用户相关消息数据的获取

关于用户发送的消息数据的获取，我们拟采用调用官方接口的方式来实现。以新浪微博为例，新浪微博对外提供了接口，用户提供授权后便可以通过代码获取到该用户的消息数据，方便开发者对微博数据的获取。但是该接口对消息获取的数量有限制，每天能够获取消息的用户数量也有限制。这与我们大数据分析的实现方案完全不符。

针对以上问题，我们进行了相关研究，得出了相应的解决方案。对于微博数量获取限制的问题，我们拟采用爬虫技术来弥补其不足。通过爬虫技术来实现微博数据的获取将不会有数量上的限制，一定程度上解决了官方接口对于消息数量获取有限制的问题。

（2）对大数据的挖掘和分析

关于消息数据的挖掘和分析，我们采用了中科院中文分词系统来实现对消息数据的挖掘，通过研究相关论文来实现对消息数据的分析。目前我们已经实现了对消息数据的挖掘和分析，但准确度还有待提高。

针对以上问题，我们进行了相关调研，得出了相应的解决方案。我们通过对中文分词系统的进一步优化和分析，将程度副词以及情感和评价类词汇再进一步进行详细划分，并且多研究一些论文来对当前算法进行优化，最终得出的结果准确度在一定程度上能够充分满足用户的需求。

五、团队介绍和人员分工

5.1 团队介绍

HelloWorld团队是一个项目开发经验丰富、充满活力的团队，团队成员共五人，全部来自于北京邮电大学，是软件学院的5名13级本科生。我们项目开发经验丰富，曾参与多个项目的开发，包括基于云计算的3D多人雕刻系统、eMusic app以及若干网站，曾在多个同类大学生软件创新大赛取得优异成绩。

5.2 人员分工

朱明晖：总体协调，情绪分析算法的设计，情感词典的构建，需求、概要设计文档的编写。

文峰：数据的获取，服务器、app的开发，情绪分析算法的设计，情感词典的构建，详细设计文档的编写。

阎硕：网页端数据管理系统的开发，情绪分析算法的设计，情感词典的构建，详细设计文档的编写。

张家杰：网页端数据管理系统的开发、测试及测试文档的编写。

暴智伟：前端界面的设计及美化，数据库的管理，需求文档的编写。

六、其他

6.1 开发结束后的感触

本项目的开发从提出想法到现在一共经过了一年的时间。目前功能已经基本实现，项目的整体开发也基本上告一段落。项目开发期间，我们遇到了许多问题。从一开始的微博数据获取失败到最后的界面设计不尽人意，每一个时间段我们似乎都有各种各样的问题出现。但也正是这些问题的出现，使得我们的应用一步一步地走向巅峰。经过此次开发，我们总结出了几点经验：

1. 项目开发初期的构思很重要，一定要将整体构思设计好，不能空有一个想法而不知道如何实现，这样的话后期开发一定会遇到诸多问题。与其后期时间严重不足，不如初期多耽误一些时间。
2. 对于界面的设计，很多时候不能空想，要试着将脑海中所想的画出来试试看。很多时候人们想的和眼睛看到的并不一样，也许一开始你觉得自己的想法很烂，但画出来之后就会觉得其实没那么烂，甚至还很有独特之处。很多新奇的想法都是这样产生出来的。
3. 项目开发过程中遇到难题要多查资料，不要自己闷在那里想。专业人士的力量肯定会强过自己，在找资料的过程中也经常会有想法不断出现，这样会比光想强很多。

此次开发获益良多，我们会总结这次经历带给我们的经验和教训，不断完善自己。

6.2 不足之处和今后设想

项目开发进行到目前这个阶段，已经基本达到了我们起初设定的目标。但不足之处依然存在，总结为以下两点：

1. 数据分析算法的准确性

为了尽可能提高算法的准确性，我们采用了大数据分析算法，通过大量的数据分析得出一个趋于准确的结果。大数据分析无疑令我们的算法准确度大幅度提升，但这并不意味着100%准确，算法准确度仍有提升空间存在。

1. 学生心理状态展现单一

本应用通过对学生在社交平台上发布的消息进行挖掘与分析，最终得出该学生的心理状态。我们成功地实现了这一关键技术，但对用户心理状态的展现还不够，心理状态还应该被进一步细化。

根据以上两点不足，我们今后的设想为：

1. 继续完善以上两点不足之处，提高算法的准确性，同时将学生的心理状态更加全面地展现出来，让高校的心理咨询师以及教育界工作者和研究人员能够更好地了解学生的心理，实时掌握学生的心理状态变化并对即将出台的政策做出相应调整。
2. 扩大该应用的用户面，将现在的面向教育界相关人士拓展成面向政府、面向公司、面向教育界和心理学相关人士的用户心理分析器。拉近政府与民众、领导与下属、老师与学生之间的距离，促进上级对下级的理解，让人们的学习、工作、生活变得更加美好。

6.3 用到的第三方代码或组件

ICTCLAS2015：中科院中文分词系统

HowNet情感词词典

七、致谢

首先，我们要感谢北京邮电大学软件学院为我们提供了良好的环境，感谢指导教师——牛琨老师和张舒博老师这些天以来对我们的悉心教导；

其次，我们要感谢用友网络iUAP中心对我们提供的技术支持。我们遇到的很多技术问题都因此而得到解决；

|  |  |
| --- | --- |
| **项目编号** | **209** |
| **文档编号** | **001** |

**心窗密语**

**需求规格说明书**

**V1.0**

**朱明晖**

2015年12月10日

目 录

[一、引言 18](#_Toc451184716)

[1.1编写目的 18](#_Toc451184717)

[1.2背景 18](#_Toc451184718)

[1.3定义 18](#_Toc451184719)

[1.4参考资料 18](#_Toc451184720)

[二、任务概述 19](#_Toc451184721)

[2.1目标 19](#_Toc451184722)

[2.2用户的特点 19](#_Toc451184723)

[2.3假定和约束 19](#_Toc451184724)

[三、需求规定 20](#_Toc451184725)

[3.1语境分析 20](#_Toc451184726)

[3.1.1周境图 20](#_Toc451184727)

[3.1.2第一级数据流图 20](#_Toc451184728)

[3.1.2第二级数据流图 21](#_Toc451184729)

[3.2功能需求 21](#_Toc451184730)

[3.2.1功能结构 21](#_Toc451184731)

[3.2.2功能描述 22](#_Toc451184732)

[3.2.3用例 23](#_Toc451184733)

[3.3非功能需求 24](#_Toc451184734)

[3.3.1 性能 24](#_Toc451184735)

[3.3.2 可靠性 24](#_Toc451184736)

[3.3.3 可用性 24](#_Toc451184737)

[3.3.4 安全 24](#_Toc451184738)

[3.3.5 可维护性 24](#_Toc451184739)

[3.3.6 可移植性 24](#_Toc451184740)

[3.4输人输出要求 24](#_Toc451184741)

[3.5逻辑数据库要求 25](#_Toc451184742)

[3.5.1 数据字典 25](#_Toc451184743)

[3.6故障处理要求 25](#_Toc451184744)

[四、运行环境规定 26](#_Toc451184745)

[4.1用户界面 26](#_Toc451184746)

[4.2硬件接口 26](#_Toc451184747)

[4.3软件接口 26](#_Toc451184748)

# 一、引言

## 1.1编写目的

本指南用于指导小组成员对“心窗”项目的开发过程，通过规范软件的开发过程达到提高软件质量，降低维护成本的目的。开发者应根据本指南进行软件开发和编制软件开发文档。预期主要读者为小组成员、指导老师及评审老师。

## 1.2背景

本项目的名称为“心窗”，作为由5名北京邮电大学13级本科生组成的小组参加大学生创新训练的项目，面向的目标用户群体为高校中的教育工作者、研究人员、心理医师及信息管理人员，用于对大学生的心理状况进行监测和控制。

## 1.3定义

IE：Internet Explorer，Internet浏览器，通过它可以访问国际互联网上的资源。

DFD：Data Flow Diagram数据流图，它从数据传递和加工角度，以图形方式来表达系统的逻辑功能、数据在系统内部的逻辑流向和逻辑变换过程。

IPO：Input, Processing, Output即输入、处理、输出表的形式。

UID：user ID微博账号。

EXCEL：Microsoft Office Excel，用于进行数据的处理、统计分析和辅助决策操作。

## 1.4参考资料

[1] 《软件工程化》，王安生，清华大学出版社。

# 二、任务概述

## 2.1目标

据相关调查显示，我国目前有1600万精神疾病患者，而青年大学生是主要群体之一。在大学生群体中，16.0%-25.4%的大学生存在不同程度的心理问题，其中约有2.8%存在严重的心理疾病。在本科生中因心理疾病休学的人数占因病休学总人数的37.9%，因心理疾病退学的人数占因病退学总人数的64.4%，均居首位。在如今青年大学生心理健康问题层出不穷的情况下，我们小组决定做一款应用来有效地对大学生心理状态进行预测和控制，避免由于严重心理问题引发的恶性案件的发生，让大学生在良好的心理状况下茁壮成长，成为社会需要的人才。本应用的目标是：脱离传统式的心理咨询模式，与互联网社交平台紧密结合，通过关注学生的微博，分析学生近期的心理状态变化，找出那些可能存在潜在心理问题的同学，帮助高校的心理医师或教育界工作者和研究人员了解学生们的心理状态、及时地关注到心理状态异动的同学。本软件产品是一项独立的软件，全部内容自含。

## 2.2用户的特点

本应用面向的最终用户是高校中的教育工作者、研究人员、心理医师及信息管理人员。教育工作者、研究人员及心理医师可以查看学生的情绪状态，只需要能够进行一些简单的手机操作，并且能够看懂情绪状态的图表。信息管理人员负责对学生、教师信息以及权限进行管理，需要熟悉计算机的操作，以及学校的管理层次。

## 2.3假定和约束

在2016年1月20日之前完成本软件的开发。

# 三、需求规定

## 3.1语境分析

### 3.1.1周境图



图1 “心窗”的周境图

### 3.1.2第一级数据流图

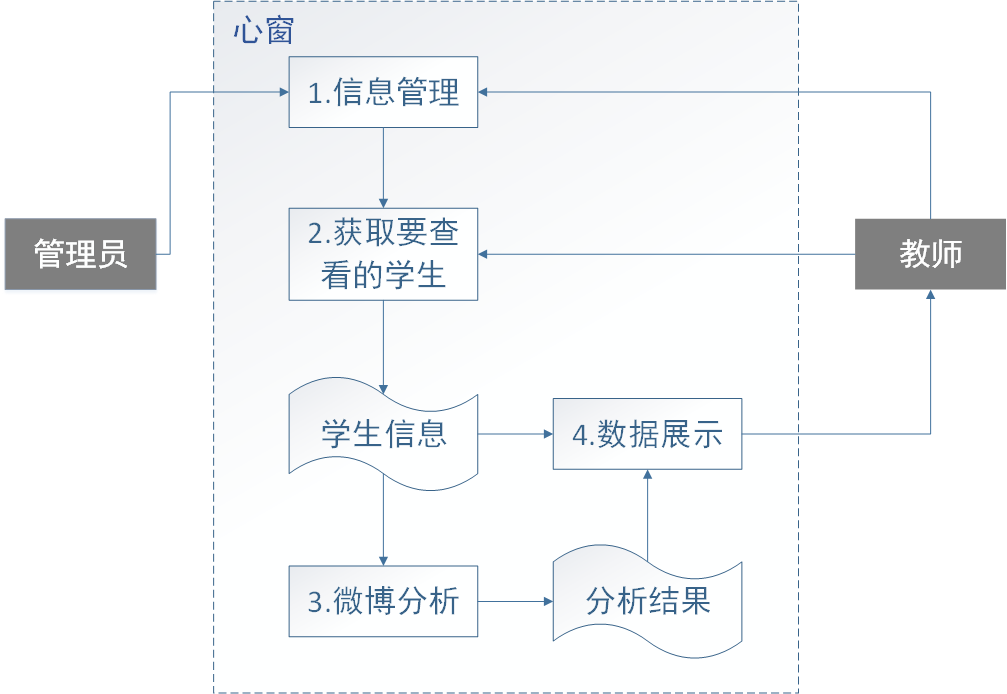


图2 用DFD表示的“心窗”初步构想

### 3.1.2第二级数据流图

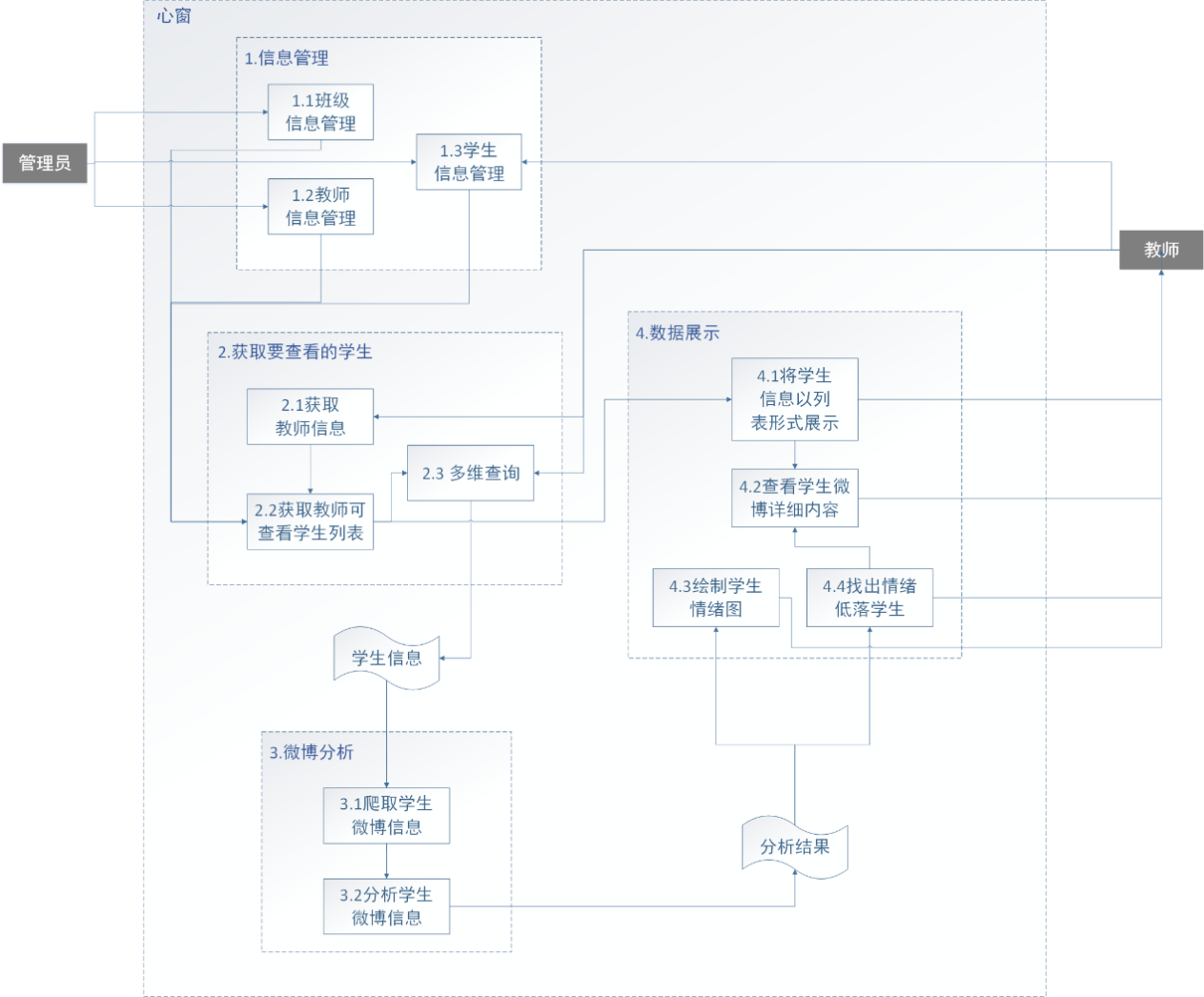


图3 用DFD表示的“心窗”较详细的模型

### 3.2功能需求

### 3.2.1功能结构

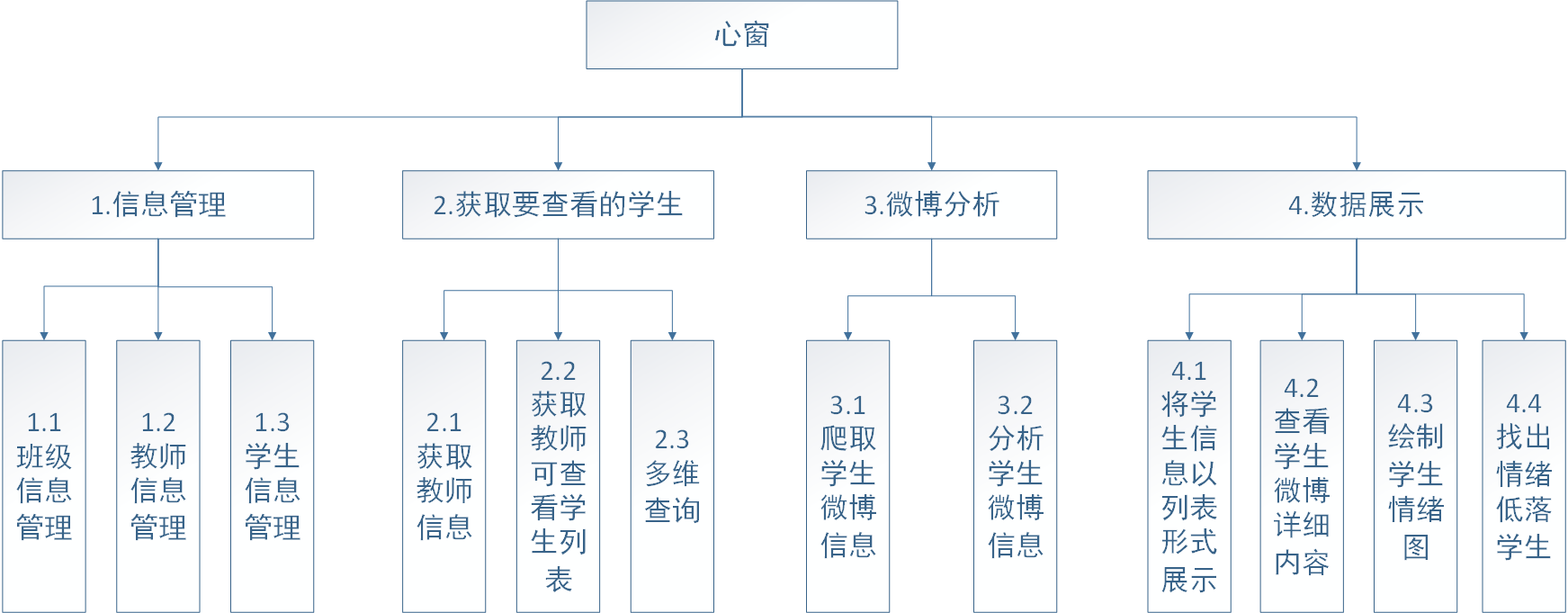


图4 “心窗”的功能结构

### 3.2.2功能描述

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 概述 | 输入信息 | 处理操作 | 输出信息 | 异常处理 |
| 1.1班级信息管理 | 对班级信息进行管理 | 班号、管理教师等 | 录入、增、删、改、查 | 相关的信息界面 | 录入时导入文件格式错误、重复录入、增删改操作错误 |
| 1.2教师信息管理 | 对教师信息进行管理 | 教师工号、姓名、职位等 | 录入、增、删、改、查 | 相关的信息界面 | 录入时导入文件格式错误、重复录入、增删改操作错误 |
| 1.3学生信息管理 | 对教师信息进行管理 | 学生学号、姓名、性别等 | 录入、增、删、改、查 | 相关信息界面 | 录入时导入文件格式错误、重复录入、增删改操作错误 |
| 2.1获取教师信息 | 获取在手机端登录的教师的信息 | 教师工号 | 无 | 教师工号 | 无 |
| 2.2获取教师可查看学生列表 | 获取该教师可查看的学生列表 | 教师工号、数据库中的班级信息表、教师信息表、学生信息表 | 数据库操作 | 可查看的学生列表 | 未联网 |
| 2.3多维查询 | 根据学生的不同属性多维选择要查看的学生 | 要查看的属性及其值 | 数据库操作 | 要查看的学生列表 | 不存在属性、不存在对应属性值 |
| 3.1爬取学生微博信息 | 根据学生的UID定向爬取学生的微博信息 | 学生UID | 爬虫 | 学生的微博信息 | 未联网 |
| 3.2分析学生微博信息 | 根据学生的微博信息分析出该微博的情绪 | 学生微博信息 | 去噪声、分词、倾向性分析 | 该微博的情绪指数 | 学生微博中的词不在词典当中 |
| 4.1学生信息展示 | 将学生信息以列表形式展示 | 可查看的学生列表 | 查看 | 相关信息界面 | 未联网 |
| 4.2查看学生微博详细内容 | 进入学生微博查看详细内容 | 学生的UID | 查看 | 相关信息界面 | 未联网 |
| 4.3绘制学生情绪图 | 根据对近期学生发布的微博的分析结果绘制情绪图 | 近期连续多条学生微博的情绪指数 | 查看 | 情绪分布图及情绪走势图 | 无 |
| 4.4找出情绪低落学生 | 当学生近期微博持续显示为低落时，将学生加入情绪低落学生列表 | 近期连续多条学生微博的情绪指数 | 查看 | 情绪低落学生列表 | 无 |

表1 “心窗”的功能模块IPO表

### 3.2.3用例



图5 “心窗”用户角色和系统功能之间的关联构成用例图

## 3.3非功能需求

### 3.3.1 性能

应用应当能够满足至少100人的同时访问，其响应速度不小于1秒；

应用在更新处理时，数据库与系统的同步误差不大于1秒；

数据的转换和传送时间不低于0.5秒；

### 3.3.2 可靠性

应用在没有发生非人为（自然灾害，断电等不可抗拒的事件）操作失误时，不得出现系统崩溃不可用的情况。

### 3.3.3 可用性

应用适合高校教师、管理员的使用。

### 3.3.4 安全

应用应该具有极高的安全性，管理员和教师只有登录了之后才能进行操作；

教师只能查看自己有权限查看的学生的学习成绩，或对其信息进行修改；

具有一定的抵御网络攻击的能力；

数据应该经常进行备份，防止丢失。

### 3.3.5 可维护性

应用应该具有向后兼容的能力；

发生系统崩溃后，能够在三天内使系统重新运行。

### 3.3.6 可移植性

本应用的移动终端版本能够在安卓4.4及以上版本系统上正常运行。浏览器版本应该能够在Windows XP以及更高的版本系统的IE6.0及以上浏览器上正常运行。

## 3.4输人输出要求

管理员录入教师信息、学生信息时，可以选择导入EXCEL表格的方式。需要导入两个表，一个是教师信息表，包括教工号、教工姓名、职位、所属院、所管班级等信息；另一个是学生信息表，包括学生学号、学生姓名、微博账号、性别、所属省份、所属班级等信息。

## 3.5逻辑数据库要求

### 3.5.1 数据字典

|  |
| --- |
| 名字：管理员信息表  描述：管理员进入应用网页端的信息  定义：管理员信息表=学校ID+学校名称+密码  位置：保存在用户信息数据库中 |
| 名字：教工信息表  描述：教工进入应用手机端的信息  定义：教工信息表=教工号+教工姓名+职位+所属学院+所管班级  位置：保存在用户信息数据库中 |
| 名字：学生信息表  描述：存储学生信息  定义：学生信息表=学生学号+学生姓名+微博账号+性别+省份+所属班级+所属学院  位置：保存在用户信息数据库中 |

表2 “心窗”的数据字典

## 3.6故障处理要求

正常使用时不应出错，若运行时遇到不可恢复的系统错误，也必须保证数据库完好无损。调试中遇到的问题及解决的方案：

1)遇到跳出“数据库已经关闭”提示信息阻止程序运行时

可以查看一下进行此项操作时，操作的表是否已经被关闭了或者是在没有关闭此表的情况下又一次运用打开语句打开此表。

2)关于空记录带来的麻烦

有些空记录往往会使程序无法运行。此时开发组成员可用“if not is null”语句先判断一下是否为空记录，再操作。

3）有些运行错误也可用捕获异常进行处理。

# 四、运行环境规定

## 4.1用户界面

本应用的用户界面友好简洁，操作简便，功能跳转自然、风格统一。

## 4.2硬件接口

本软件不需要特定的硬件或硬件接口进行支撑。

586以上PC机均可运行此软件。

## 4.3软件接口

本应用移动终端版本仅可运行于安卓4.4及以上版本的系统。应用大小可根据屏幕大小自动适配。

本应用浏览器版本运行于Windows95及更高版本的Windows操作系统、IE6.0及其兼容版本。

|  |  |
| --- | --- |
| **项目编号** | **209** |
| **文档编号** | **002** |

**心窗密语**

**概要设计说明书**

**V1.0**

**朱明晖**

2015年12月20日

目 录

[一、引言 29](#_Toc451185718)

[1.1 编写目的 29](#_Toc451185719)

[1.2 背景 29](#_Toc451185720)

[1.3 定义 29](#_Toc451185721)

[1.4 参考资料 29](#_Toc451185722)

[二、设计概述 30](#_Toc451185723)

[2.1限制和约束 30](#_Toc451185724)

[2.2设计原则和设计要求 30](#_Toc451185725)

[2.2.1命名规则 30](#_Toc451185726)

[2.2.2模块独立性原则 30](#_Toc451185727)

[2.2.3设计要求 30](#_Toc451185728)

[三、系统逻辑设计 30](#_Toc451185729)

[3.1系统组织设计 30](#_Toc451185730)

[3.2系统结构设计 31](#_Toc451185731)

[3.3系统接口设计 34](#_Toc451185732)

[3.3.1系统接口表 34](#_Toc451185733)

[四、系统出错处理设计 35](#_Toc451185734)

[4.1系统出错处理表 35](#_Toc451185735)

[五、技术设计 37](#_Toc451185736)

[5.1系统开发技术说明表 37](#_Toc451185737)

[六、数据库设计 37](#_Toc451185738)

[6.1逻辑设计 37](#_Toc451185739)

[6.2物理设计 38](#_Toc451185740)

[6.2.1表汇总 名称简写 38](#_Toc451185741)

[6.2.2表详情 38](#_Toc451185742)

# 一、引言

## 编写目的

本指南用于指导小组成员对“心窗”项目的开发过程，基于其软件需求规格说明书进行编写，编写此说明书的主要目的有以下几点：

1、对“心窗”项目概要设计的阶段任务成果形成文档，为阶段以及最终的验收、评审提供便利。

2、对需求阶段的文档再次确认过程，对前一阶段需求没有做充分或错误的提出修改。

3、明确整个系统的功能框架和数据库结构，为下一阶段的详细设计、编码、和测试提供参考依据。

4、明确编码规范和命名规范，统一程序界面。

预期主要读者为小组成员、指导老师及评审老师。

## 背景

本项目的名称为“心窗”，作为由5名北京邮电大学13级本科生组成的小组参加大学生创新训练的项目，面向的目标用户群体为高校中的教育工作者、研究人员、心理医师及信息管理人员，用于对大学生的心理状况进行监测和控制。

## 定义

UID：user ID微博账号。

EXCEL：Microsoft Office Excel，用于进行数据的处理、统计分析和辅助决策操作。

## 参考资料

[1] 本项目的软件需求规格说明书；

[2] 《软件工程化》，王安生，清华大学出版社。

# 二、设计概述

## 2.1限制和约束

在2016年4月之前完成本软件的开发。

## 2.2设计原则和设计要求

### 2.2.1命名规则

在本软件开发过程中，一律按照骆驼式命名法（Camel-Case）进行命名。

### 2.2.2模块独立性原则

模块功能要单一；新增功能时，尽量减少对已有模块的改动，在新增模块中实现功能；功能调整或修改bug时，影响的模块数量要尽可能少；功能调整或修改bug时，受影响的模块要尽可能的小；模块间要松耦合，模块接口尽可能少、并且简单。

### 2.2.3设计要求

有关对本应用的安全性、灵活性、易操作性、可维护性要求，参见本应用的需求规格说明书。

# 三、系统逻辑设计

## 3.1系统组织设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | **英文名称** | **中文名称** | **业务职能** | **安装地点** | **备注** |
| 1 | Server | 服务器 | 数据传输及学生微博情绪分析等 | 部署在云服务器上 | 无 |
| 2 | Web | 客户端网页端 | 信息批量录入及管理 | 部署在云服务器上 | 仅允许高校管理员使用 |
| 3 | App | 客户端手机端 | 学生情绪展示及信息管理 | 教工的手机上 | 根据老师的工号确定老师可以查看的学生 |

表1 “心窗”的系统组织表

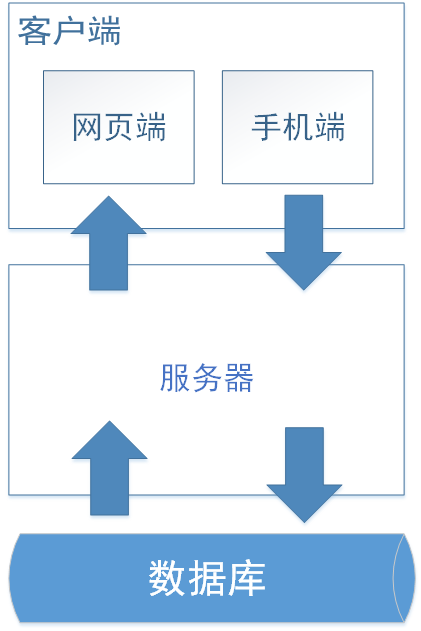


图1 “心窗”的总体系统组织图

## 3.2系统结构设计

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 1 | | | | |
| **子系统英文名称** | | Server | | | | |
| **子系统中文名称** | | 服务器 | | | | |
| **特性编号** | **系统特征**  **英文名称** | **系统特征**  **中文名称** | **操作功能** | **调用对象** | **被调用**  **对象** | **备注** |
| 1 | Data Transmission | 数据传输 | 与客户端和数据库之间进行数据传输 | 系统特性3、4、5、6、7、8、9 | 数据库 | 无 |
| 2 | Emotion Analysis | 情绪分析 | 获取学生近期微博信息并进行分析，绘制出情绪图 | 系统特性8、10 | 无 | 必须有学生的UID |
| 说明：用于将本地数据库与客户端相连；通过学生的UID对其微博内容进行分析，绘制出情绪图，展示在客户端的手机端上。 | | | | | | |

表2 “心窗”的服务器子系统的系统特性表

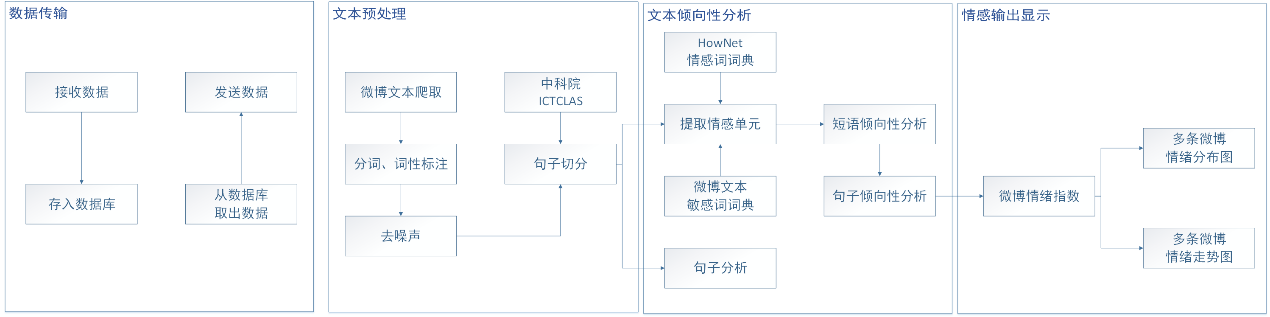


图2 “心窗”的服务器子系统的系统特性结构图

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 2 | | | | |
| **子系统英文名称** | | Web | | | | |
| **子系统中文名称** | | 客户端网页端 | | | | |
| **特性编号** | **系统特征**  **英文名称** | **系统特征**  **中文名称** | **操作功能** | **调用对象** | **被调用**  **对象** | **备注** |
| 3 | Register | 注册 | 注册一个管理员 | 操作界面 | 系统特性1：数据传输 | 需要填写所需信息 |
| 4 | Login | 登录 | 用管理员账号密码进行登录 | 操作界面 | 系统特性1：数据传输 | 注意错误处理 |
| 5 | Query | 查看信息 | 查看本校的学院、班级、教师、学生信息 | 操作界面 | 系统特性1：数据传输 | 无 |
| 6 | Modify | 修改信息 | 对本校学院、班级、教师、学生信息进行增、删、改操作 | 操作界面 | 系统特性1：数据传输 | 无 |
| 说明：管理员登录，对本校学生及教工的信息进行录入和管理。 | | | | | | |

表3 “心窗”的客户端网页端子系统的系统特性表

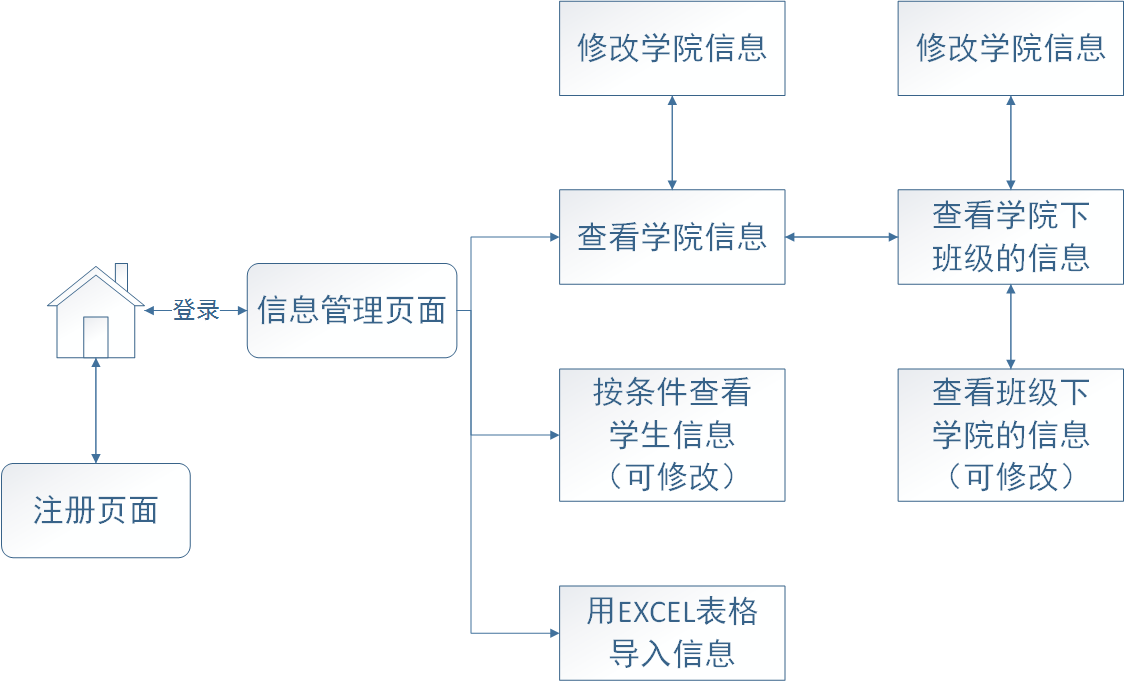


图3 “心窗”的客户端网页端子系统的系统特性结构图

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 3 | | | | |
| **子系统英文名称** | | App | | | | |
| **子系统中文名称** | | 客户端手机端 | | | | |
| **特性编号** | **系统特征**  **英文名称** | **系统特征**  **中文名称** | **操作功能** | **调用对象** | **被调用**  **对象** | **备注** |
| 7 | Login | 登录 | 教工选择学校后以教工号进行登录 | 操作界面 | 系统特性1：数据传输 | 无 |
| 8 | Check Student Emotion | 查看学生情绪 | 查看指定学生的情绪图（分布图及走势图） | 操作界面 | 系统特性1：数据传输、系统特性2：情绪分析 | 可以查看全校学生整体情绪、本班（院）学生整体情绪、具体学生个人情绪，还可以按条件查看学生的情绪 |
| 9 | Check Student List | 查看学生列表 | 查看学生的详细信息 | 操作界面 | 系统特性1：数据传输 | 可以查看情绪低落学生列表 |
| 10 | Directional Search | 定向搜索 | 根据微博号定向搜索学生的情绪 | 操作界面 | 系统特性2：情绪分析 | 也可选择提供的热门微博查看其情绪 |
| 11 | Enter Blog | 进入学生微博 | 进入学生微博查看详情 | 操作界面 | 无 | 无 |
| 说明：教工登录，可以看到全校学生的整体情绪图及有权限查看的学生的情绪图，还可以按条件对学生进行查看。同时也可以对学生的信息进行管理。 | | | | | | |

表4 “心窗”的客户端手机端子系统的系统特性表

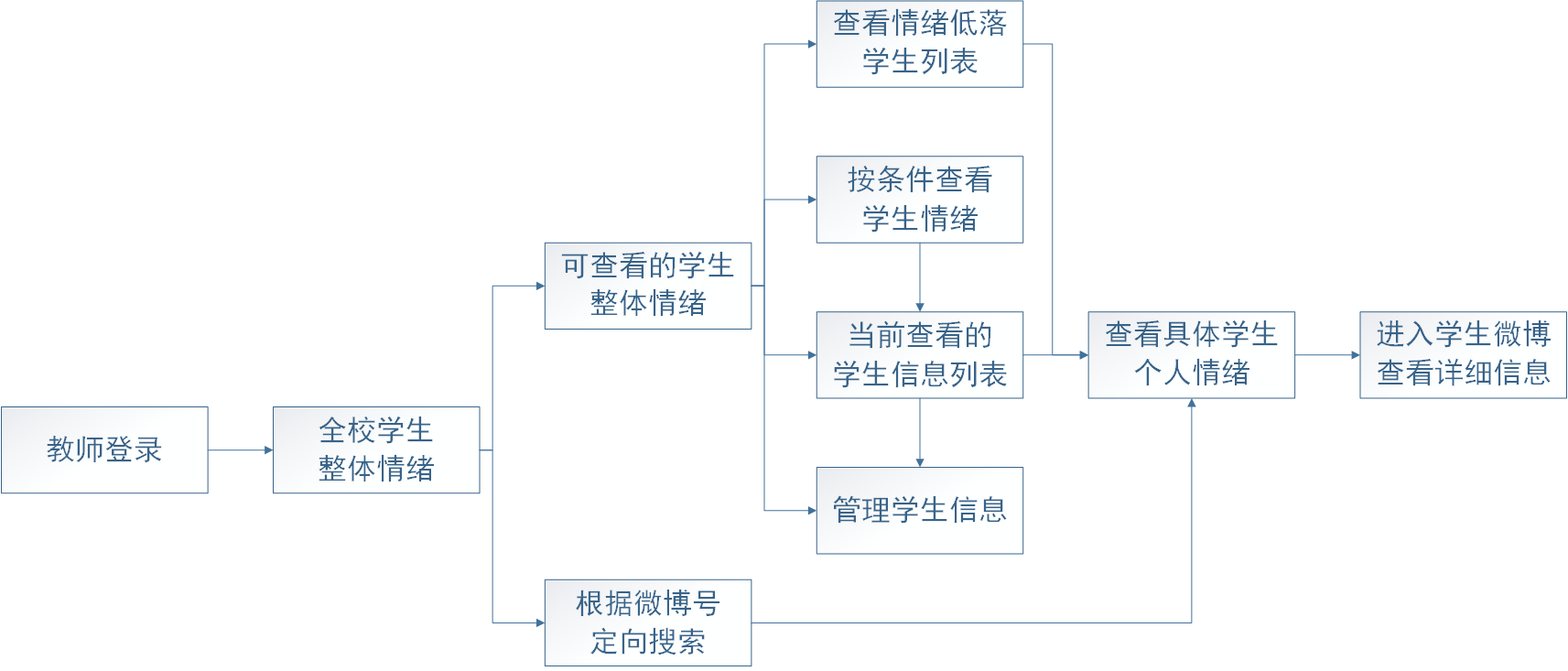


图4 “心窗”的客户端手机端子系统的系统特性结构图

## 3.3系统接口设计

### 3.3.1系统接口表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 1 | | | |
| **子系统英文名称** | | Server | | | |
| **子系统中文名称** | | 服务器 | | | |
| **接口编号** | **接口名称** | **接口功能** | **接口类型** | **接口性质** | **备注** |
| 1 | 数据传输接口 | 调用系统特性1：数据传输，实现服务器与客户端和数据库的数据传输功能 | 内部接口 | 双向 | 无 |
| 2 | 情绪分析 | 调用系统特性2：情绪分析，爬取学生微博，分析其情绪并绘制情绪图 | 内部接口 | 输出 | 无 |
| 说明：各系统特性通过这两个接口调用服务器子系统的系统特性 | | | | | |

表5 “心窗”的服务器子系统的系统接口表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 2 | | | |
| **子系统英文名称** | | Web | | | |
| **子系统中文名称** | | 客户端网页端 | | | |
| **接口编号** | **接口名称** | **接口功能** | **接口类型** | **接口性质** | **备注** |
| 3 | 网页用户界面 | 调用系统特性3、4、5、6，进行信息的录入及管理 | 用户接口 | 双向 | 仅管理员可使用 |
| 说明：管理员操作用户界面进行信息的录入及管理 | | | | | |

表6 “心窗”的客户端网页端子系统的系统接口表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 3 | | | |
| **子系统英文名称** | | App | | | |
| **子系统中文名称** | | 客户端手机端 | | | |
| **接口编号** | **接口名称** | **接口功能** | **接口类型** | **接口性质** | **备注** |
| 4 | 应用用户界面 | 调用系统特性7、8、9、10、11，进行对学生情绪的查看和管理 | 用户接口 | 双向 | 教工登录使用 |
| 说明：教工登录进行对学生情绪的查看和管理 | | | | | |

表7“心窗”的客户端手机端子系统的系统接口表

# 四、系统出错处理设计

## 4.1系统出错处理表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 1 | | | |
| **子系统英文名称** | | Server | | | |
| **子系统中文名称** | | 服务器 | | | |
| **错误编号** | **错误名称** | **错误原因** | **错误信息** | **处理方式** | **备注** |
| 1 | 连接失败 | 服务器与客户端连接失败或与数据库连接失败 | 服务器端连接出现问题，请稍后重试 | 返回信息 | 无 |
| 2 | 异常值 | 值越界、不存在等情况 | 无 | 用try/catch  处理异常 | 无 |
| 说明：编程时要做详尽的错误处理，保证服务器不至于无法运行。 | | | | | |

表8 “心窗”的服务器子系统的系统出错处理表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 2 | | | |
| **子系统英文名称** | | Web | | | |
| **子系统中文名称** | | 客户端网页端 | | | |
| **错误编号** | **错误名称** | **错误原因** | **错误信息** | **处理方式** | **备注** |
| 3 | 连接失败 | 客户端与服务器连接失败 | 与服务器连接失败，请确保网络连接后再重试 | 返回信息 | 无 |
| 4 | 异常值 | 值越界、不存在等情况 | 无 | 用try/catch  处理异常 | 无 |
| 说明：编程时要做详尽的错误处理，保证客户端网页端不至于无法运行。 | | | | | |

表9 “心窗”的客户端网页端子系统的系统出错处理表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 3 | | | |
| **子系统英文名称** | | App | | | |
| **子系统中文名称** | | 客户端手机端 | | | |
| **错误编号** | **错误名称** | **错误原因** | **错误信息** | **处理方式** | **备注** |
| 5 | 连接失败 | 客户端与服务器连接失败 | 与服务器连接失败，请确保网络连接后再重试 | 返回信息 | 无 |
| 6 | 异常值 | 值越界、不存在等情况 | 无 | 用try/catch  处理异常 | 无 |
| 说明：编程时要做详尽的错误处理，保证客户端手机端不至于无法运行。 | | | | | |

表10 “心窗”的客户端网页端子系统的系统出错处理表

# 五、技术设计

## 5.1系统开发技术说明表

在此，一些非常常用的开发技术不作冗述（例如JAVA开发，C++开发），只列出一些不常见的技术。如下面的表所示。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **子系统编号** | | 1 | | | |
| **子系统英文名称** | | Server | | | |
| **子系统中文名称** | | 服务器 | | | |
| **技术编号** | **开发技术**  **英文名称** | **开发技术**  **中文名称** | **处理功能** | **系统特征编号** | **备注** |
| 1 | ICTCLAS | 中科院汉语词法分析系统 | 对文本信息进行分词、词性标注、去噪声、句子切分等功能 | 2 | 无 |
| 说明： | | | | | |

表11 “心窗”的服务器子系统的系统开发技术说明表

# 六、数据库设计

## 6.1逻辑设计

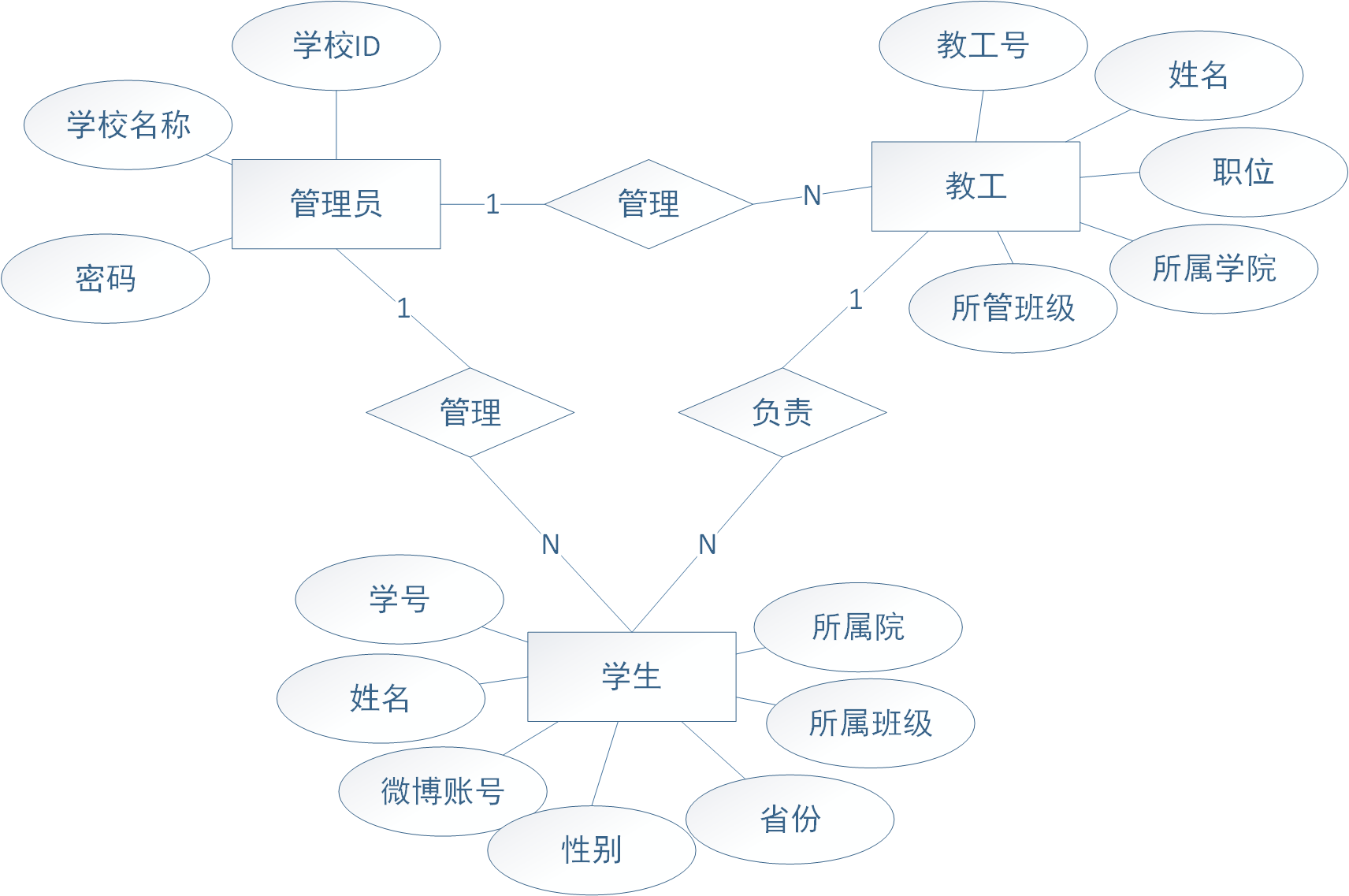


图5 “心窗”的数据库E-R图

## 6.2物理设计

### 6.2.1表汇总

|  |  |
| --- | --- |
| **表名称** | **功能说明** |
| university | 用于记录管理员的信息 |
| universityID\_teacher | 用于记录学校ID为universityID的学校的教师信息 |
| universityID\_student | 用于记录学校ID为universityID的学校的学生信息 |

表12 “心窗”的数据库表汇总

### 6.2.2表详情

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名称** | | university | | | | | |
| **数据库用户** | | 系统维护人员 | | | | | |
| **主键** | | universityID | | | | | |
| **其他排序字段** | | unversityName, password | | | | | |
| **序号** | **字段名称** | **数据类型** | **允许为空** | **唯一** | **区别度** | **默认值** | **约束/说明** |
| 1 | universityID | int | N | Y |  |  |  |
| 2 | universityName | char(50) | N | Y |  |  |  |
| 3 | password | char(50) | N | N |  |  |  |
| **记录数** | |  | | | | | |
| **增长量** | |  | | | | | |
| **表的并发** | |  | | | | | |
| **补充说明** | |  | | | | | |

表13 university表详细设计表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名称** | | universityID\_teacher | | | | | |
| **数据库用户** | | 管理员 | | | | | |
| **主键** | | teacherID | | | | | |
| **其他排序字段** | | teacherName, occupation, academy, classID | | | | | |
| **序号** | **字段名称** | **数据类型** | **允许为空** | **唯一** | **区别度** | **默认值** | **约束/说明** |
| 1 | teacherID | int | N | Y |  |  |  |
| 2 | teacherName | char(50) | N | N |  |  |  |
| 3 | occupation | char(50) | N | N |  |  |  |
| 4 | academy | char(50) | N | N |  |  |  |
| 5 | classID | int | Y | N |  |  |  |
| **记录数** | |  | | | | | |
| **增长量** | |  | | | | | |
| **表的并发** | |  | | | | | |
| **补充说明** | |  | | | | | |

表14 universityID\_teacher表详细设计表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名称** | | universityID\_student | | | | | |
| **数据库用户** | | 管理员 | | | | | |
| **主键** | | studentID | | | | | |
| **其他排序字段** | | studentName, UID, sex,  province, classID, grade, academy | | | | | |
| **序号** | **字段名称** | **数据类型** | **允许为空** | **唯一** | **区别度** | **默认值** | **约束/说明** |
| 1 | studentID | int | N | Y |  |  |  |
| 2 | studentName | char(50) | N | N |  |  |  |
| 3 | UID | int | N | Y |  |  |  |
| 4 | sex | char(50) | N | N |  |  |  |
| 5 | province | char(50) | N | N |  |  |  |
| 6 | classID | int | N | N |  |  |  |
| 7 | grade | char(50) | N | N |  |  |  |
| 8 | academy | char(50) | N | N |  |  |  |
| **记录数** | |  | | | | | |
| **增长量** | |  | | | | | |
| **表的并发** | |  | | | | | |
| **补充说明** | |  | | | | | |

表15 universityID\_student表详细设计表

|  |  |
| --- | --- |
| **项目编号** | **209** |
| **文档编号** | **003** |

**心窗密语**

**详细设计说明书**

**V1.0**

**阎硕、文峰**

2015年1月2日

**目 录**

[一、引言 43](#_Toc451185256)

[二、支撑环境 44](#_Toc451185257)

[2.2开发工具、中间件以及数据库接口 44](#_Toc451185258)

[2.2.1开发工具 44](#_Toc451185259)

[三、部件详细设计 44](#_Toc451185260)

# 一、引言

## 1.1编写目的

本指南用于指导小组成员对“心窗”项目的开发过程，基于其软件需求规格说明书及概要设计说明书进行编写。本详细设计说明书在对功能需求进行确认的基础上，设计子系统的详细结构模型、详细信息模型，并详细描述各主要程序模块的处理逻辑、各输入／输出格式及基础数据的编码方案。它是软件实施的唯一依据，也是将来系统维护的主要参考资料。

预期主要读者为小组成员、指导老师及评审老师。

## 1.2背景

本项目的名称为“心窗”，作为由5名北京邮电大学13级本科生组成的小组参加大学生创新训练的项目，面向的目标用户群体为高校中的教育工作者、研究人员、心理医师及信息管理人员，用于对大学生的心理状况进行监测和控制。

## 1.3定义

UID：user ID微博账号。

EXCEL：Microsoft Office Excel，用于进行数据的处理、统计分析和辅助决策操作。

ID: identification.

## 1.4参考资料

[1] 本项目的软件需求规格说明书；

[2] 本项目的软件概要设计说明书；

[3] 《软件工程化》，王安生，清华大学出版社。

# 二、支撑环境

## 2.1数据库管理系统

本应用使用的数据库管理系统是MySQL, 由瑞典MySQL AB公司开发，目前属于[Oracle](http://baike.baidu.com/view/15020.htm)旗下公司。版本号为Ver 14.14 Distrib 5.6.19(x86\_64)。

描述数据库管理系统、以及安装配置情况，需要描述的内容可能包括：

## 2.2开发工具、中间件以及数据库接口

## 2.2.1开发工具

本应用的客户端的网页端是用NetBeans进行开发的， 由Sun公司（2009年被[甲骨文](http://baike.baidu.com/view/8170.htm)收购）在2000年创立。版本号为8.0.2，采用JAVA EE技术中的JSF框架开发的模式，运用了JAVA、html、sql等语言。

本应用的客户端的手机端是用UAP Mobile进行开发的，由用友iUAP公司推出。版本号为2.7.20150918，运用了JavaScript和html等语言。

本应用的服务器是用Eclipse进行开发的，Eclipse最初是由[IBM](http://baike.baidu.com/subview/1937/5817096.htm)公司开发的替代[商业软件](http://baike.baidu.com/view/194971.htm)Visual Age for Java的下一代[IDE](http://baike.baidu.com/subview/5775/5401660.htm)[开发环境](http://baike.baidu.com/view/4831305.htm)，2001年11月贡献给开源社区，现在它由非营利软件供应商联盟Eclipse基金会（Eclipse Foundation）管理。运用了JAVA等语言。

# 三、部件详细设计

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 1 | 部件名称 | Data Tranmission |
| 所属子系统 | Server | | |
| 部件调用者 | 系统特性3 4 5 6 7 8 9 | | |
| 部件被调用者 | 数据库 | | |
| 部件入口参数 | Operation：需要执行的SQL操作，不同的操作给出的入口参数的数据类型也各不相同。 | | |
| 部件出口参数 | result：数据库操作的结果。根据不同的系统特性，出口参数的数据类型也不尽相同。 | | |
| 算法：  Class.forName("com.mysql.jdbc.Driver");  Connection conn=DriverManager.getConnection(数据库URL,用户名,密码);  Statement stat=conn.createStatement();  if(SELECT语句){  ResultSet rs=stat.executeQuery(SQL语句);  return 查询结果;  }else if(UPDATE语句,DELETE语句或INSERT语句){  stat.executeUpdate(SQL语句);  }else if(CREATE语句){  stat.execute(SQL语句);  } | | | |
| 流程描述：  系统通过该过程与数据库建立连接，从而进行数据查询等操作，与其他过程进行数据传输。该过程通过JDBC，使用数据库的URL，用户名和密码与MYSQL数据库建立连接。连接建立成功后，使用Statement和Resultset，以及SQL语句来执行数据库查询、更新、删除，以及建表操作。 | | | |
| 表示形式 | 过程 | 运行环境 |  |
| 性能要求 |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 3 | 部件名称 | Register |
| 所属子系统 | Web | | |
| 部件调用者 | 操作界面 | | |
| 部件被调用者 | 系统特性1：数据传输 | | |
| 部件入口参数 | String user：用户执行注册操作时输入的用户名  String password：用户执行注册操作时输入的密码  String passwordAgain：用户执行注册操作时输入的确认密码  String universityName：用户执行注册操作时输入的学校名称 | | |
| 部件出口参数 | String result：处理后得出的注册结果 | | |
| 算法：  if(user为空)  return 用户名不能为空;  else if(password为空)  return密码不能为空;  else if(passwordAgain为空)  return密码不能为空;  else if(universityName为空)  return 学校不能为空;  else{  if(password与passwordAgain相等){  if(universityName在数据库中不存在){  将注册信息加入数据库;  return 注册成功;  }else  return 该学校已经被注册;  }else  return 两次密码输入不一致;  } | | | |
| 流程描述：  系统将用户在界面执行注册操作时输入的注册数据传送至后台。首先判断用户名、密码、确认密码、学校名称四条注册信息全都不为空；然后，判断密码和确认密码是否一致；若密码一致，则再判断学校名称是否已经被注册；若该学校没有被注册，则返回结果，注册成功，其他情况均属于注册失败。 | | | |
| 表示形式 | 过程 | 运行环境 |  |
| 性能要求 |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 4 | 部件名称 | Login |
| 所属子系统 | Web | | |
| 部件调用者 | 系统特性1：数据传输 | | |
| 部件被调用者 | 操作界面 | | |
| 部件入口参数 | String user：用户执行登录操作时输入的用户名  String password：用户执行登录操作时输入的密码 | | |
| 部件出口参数 | String result： | | |
| 算法：  If(user为空)  return 用户名不能为空;  else if(password为空)  return 密码不能为空  else{  if(user在数据库中存在)  if(password与数据库中user所对应的密码相同)  return 登录成功;  else  return 密码错误;  else  return 用户名不存在;  } | | | |
| 流程描述：  系统将用户在界面执行登录操作时输入的数据传送至后台。首先判断用户名、密码、全都不为空；然后，判断该用户名在数据库中是否存在；若用户名存在，则再判断密码是否匹配；若密码匹配成功，则返回结果，登录成功，其他情况均属于登录失败。 | | | |
| 表示形式 | 过程 | 运行环境 |  |
| 性能要求 |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 5 | 部件名称 | Query |
| 所属子系统 | Web | | |
| 部件调用者 | 系统特性1：数据传输 | | |
| 部件被调用者 | 操作界面 | | |
| 部件入口参数 | String universityID：当前用户注册的学校ID  String academy：用户点击查看的学院  String cclass：用户点击查看的班级 | | |
| 部件出口参数 | List<Class> result：根据不同的层次，返回不同类型的Class类的List | | |
| 算法：  If(academy不为空){  根据academy查询该学院中所有的班级;  return list<Cclass>;  }else if(cclass不为空){  根据cclass查询该班级中所有的学生;  return list<Student>;  }else{  根据universityID查询该学校中所有的学院;  Return list<Academy>;  } | | | |
| 流程描述：  用户首先进入学院浏览界面，后台调出该学校的所有学院供用户查看；用户点击查看班级之后将该学院信息传给后台，后台调出该学院的所有班级供用户查看；用户点击查看学生之后将该班级信息传给后台，后台调出该学院的所有学生供用户查看。 | | | |
| 表示形式 | 过程 | 运行环境 |  |
| 性能要求 |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 6 | 部件名称 | Modify |
| 所属子系统 | Web | | |
| 部件调用者 | 系统特性1：数据传输 | | |
| 部件被调用者 | 操作界面 | | |
| 部件入口参数 | String old：根据不同层次、不同类型的数据，将修改之前的信息传送至后台  String new：若为修改操作，需要把修改之后的信息传送至后台 | | |
| 部件出口参数 | 无 | | |
| 算法：  switch(操作类型){  case “修改操作”:  通过old在数据库中找到需要修改的位置，并将其修改为new;  break;  case “增加操作”:  在数据库中增加一个默认条目;  break;  case “删除操作”:  通过old在数据库中找到需要删除的位置，并将其删除;  break;  } | | | |
| 流程描述：  首先判断出操作类型。若为修改操作，则使用修改前和修改后的数据对数据库进行UPDATE语句操作；若为增加操作，则直接在数据库中使用INSERT语句增加一个默认条目，然后再进行修改；若为删除操作，则使用删除前的数据对数据库进行DELETE语句操作。 | | | |
| 表示形式 | 过程 | 运行环境 |  |
| 性能要求 |  |  |  |

|  |  |
| --- | --- |
| **项目编号** | **209** |
| **文档编号** | **004** |

**心窗密语**

**测试说明书**

**V1.0**

**张家杰**

2015年1月5日

**目 录**

[一、功能测试 51](#_Toc451186483)

1.[1 Web应用的测试方法 51](#_Toc451186484)

1.[2 手机应用程序的测试方法 53](#_Toc451186485)

1.[3 链接测试 53](#_Toc451186486)

1.[4 表单测试 53](#_Toc451186487)

1.[5 设计语言测试 53](#_Toc451186488)

1.[6 数据库测试 53](#_Toc451186489)

[二、GUI基本测试内容 54](#_Toc451186490)

2.[1窗口 54](#_Toc451186491)

2.[2下拉式菜单和鼠标操作 54](#_Toc451186492)

2.[3数据项 55](#_Toc451186493)

[三、性能测试 55](#_Toc451186494)

3.[1简介 55](#_Toc451186495)

3.[2测试环境： 55](#_Toc451186496)

3.[3测试内容： 55](#_Toc451186497)

3.[4测试结果的分析说明 56](#_Toc451186498)

本项目是基于B/S和C/S两种技术开发的，用户可以通过手机APP和电脑的浏览器访问服务器。在这里着重从java开发Android和网站的角度，结合项目特点，把软件测试分为以下几部分：

# 一、功能测试

功能测试就是对产品的各功能进行验证，根据功能测试用例，逐项测试，检查产品是否达到用户要求的功能。

## 1.1 Web应用的测试方法

（1）页面链接检查： 每一个链接是否都有对应的页面，并且页面之间切换工具。

（2）相关性检查：删除/增加一项会不会对其他项产生影响，如果产生影响，这些影响是否都正确检查按钮的功能是否正确 如新建、编辑、删除、关闭、返回、保存、导入等功能是否正确。

（3）字符类型检查：在应该输入指定类型的内容的地方输入其他类型的内容(如在应该输入整型的地方，输入其他字符类型),看系统是否检查字符类型。

1）标点符号检查：输入内容包括各种标点符号,特别是空格,各种引号,回车键。看系统处理是否正确

2）特殊字符检查：输入特殊符号，如@、#、$、%、!等，看系统处理是否正确。

3)字符串长度检查: 输入超出需求所说明的字符串长度的内容, 看系统是否检查字符串长度。

（4）中文字符处理：在可以输入中、英文的系统输入中文,看会否出现乱码或出错。检查信息的完整性 在查看信息和更新信息时,查看所填写的信息是不是全部更新,更新信息和添加信息是否一致。

（5）信息重复：在一些需要命名,且名字应该唯一的信息输入重复的名字或ID,看系统有没有处理,是否会报错,重名包括是否区分大小写,以及在输入内容的前后输入空格,系统是否作出正确处理。

（6）检查删除功能：在一些可以一次删除多个信息的地方,不选择任何信息,按“删除”,看系统如何处理,是否出错;然后选择一个和多个信息,进行删除,看是否正确处理。

（7）检查添加和修改是否一致：检查添加和修改信息的要求是否一致,例如添加要求必填的项,修改也应该必填;添加规定为整型的项,修改也必须为整型

（8）检查修改重名：修改时把不能重名的项改为已存在的内容,看是否处理,报错.同时,也要注意,会不会报和自己重名的错

（9）重复提交表单：一条已经成功提交的纪录，返回后再提交，看看系统是否做了处理。对于Web系统检查多次使用返回键的情况 在有返回键的地方,返回到原来页面,重复多次，看会否出错

（10）搜索检查：有搜索功能的地方输入系统存在和不存在的内容,看搜索结果是否正确.如果可以输入多个搜索条件,可以同时添加合理和不合理的条件,看系统处理是否正确。

（11）输入信息位置：注意在光标停留的地方输入信息时,光标和所输入的信息会否跳到别的地方。

（12）上传下载文件检查：上传下载文件的功能是否实现，上传文件是否能打开。对上传文件的格式有何规定，系统是否有解释信息，并检查系统是否能够做到。下载文件能否打开或者保存，下载的文件是否有格式要求，如需要特殊工具才可以打开等。

（13）必填项检查：应该填写的项没有填写时系统是否都做了处理，对必填项是否有提示信息，如在必填项前加“\*”；对必填项提示返回后，焦点是否会自动定位到必填项。

（14）快捷键检查：是否支持常用快捷键，如Ctrl+C、 Ctrl+V、 Backspace等，对一些不允许输入信息的字段，如选人，选日期对快捷方式是否也做了限制。

（15）回车键检查：在输入结束后直接按回车键,看系统处理如何,会否报错。

（16）刷新键检查：在Web系统中，使用浏览器的刷新键，看系统处理如何，会否报错。

（17）回退键检查：在Web系统中，使用浏览器的回退键，看系统处理如何，会否报错。对于需要用户验证的系统，在退出登录后，使用回退键，看系统处理如何；多次使用回退键，多次使用前进键，看系统如何处理。

（18）直接URL链接检查：在Web系统中，直接输入各功能页面的URL地址，看系统如何处理，对于需要用户验证的系统更为重要。

（19）空格检查：在输入信息项中，输入一个或连串空格，查看系统如何处理。如对于要求输入整型、符点型变量的项中，输入空格，既不是空值，又不是标准输入。

（20）输入法半角全角检查：在输入信息项中，输入半角或全角的信息，查看系统如何处理。如对于要求输入符点型数据的项中，输入全角的小数点（“。”或“．”，如４．５）；输入全角的空格等。

（21）密码检查：一些系统的加密方法采用对字符Ascii码移位的方式，处理密码加密相对较为简单，且安全性较高，对于局域网系统来说，此种方式完全可以起到加密的作用，但同时，会造成一些问题，即大于128的Ascii对应的字符在解密时无法解析，尝试使用“uvwxyz”等一些码值较大的字符作为密码，同时，密码尽可能的长，如17位密码等，造成加密后的密码出现无法解析的字符。

（22）用户检查：任何一个系统，都有各类不同的用户，同样具有一个或多个管理员用户，检查各个管理员之间是否可以相互管理，编辑、删除管理员用户。同时，对于一般用户，尝试删除，并重建同名的用户，检查该用户其它信息是否重现。同样，提供注销功能的系统，此用户再次注册时，是否作为一个新的用户。

（23）系统数据检查：这是功能测试最重要的，如果系统数据计算不正确，那么功能测试肯定是通不过的。数据检查根据不同的系统，方法不同。对于业务管理平台，数据随业务过程、状态的变化保持正确，不能因为某个过程出现垃圾数据，也不能因为某个过程而丢失数据。

（24）系统可恢复性检查：以各种方式把系统搞瘫，测试系统是否可正常迅速恢复。

（25） 文件数据格式检查：上传的excel表格中，如果用户的数据格式错误，是否会进行检查并提示用户错误的位置，方便用户修改。

## 1.2 手机应用程序的测试方法

（1）信息查询检查：用户点击查看相关信息时，服务器是否会返回查询的信息。

（2）检查按钮的正确性：操作不同的按钮时，检查系统是否能够有相对应的事件处理，并且检查事件处理的正确性。

## 1.3 链接测试

链接是Web应用系统的一个主要特征，它是在页面之间切换和指导用户去一些不知道地址的页面的主要手段。链接测试可分为三个方面：

首先，测试所有链接是否按指示的那样确实链接到了该链接的页面；

其次，测试所链接的页面是否存在；

最后，保证Web应用系统上没有孤立的页面，所谓孤立页面是指没有链接指向该页面，只有知道正确的URL地址才能访问。

## 1.4. 表单测试

当用户给Web应用系统管理员提交信息时，就需要使用表单操作，例如用户注册、登陆、信息提交等。在这种情况下，我们必须测试提交操作的完整性，以校验提交给服务器的信息的正确性。例如：用户填写的用户名是否在系统中已存在。密码是否只能接受某些字符，测试时可以跳过这些字符，看系统是否会报错。

## 1.5 设计语言测试

Web设计语言版本的差异可以引起客户端或服务器端严重的问题，例如使用哪种版本的HTML等。当在分布式环境中开发时，开发人员都不在一起，这个问题就显得尤为重要。除了HTML的版本问题外，不同的脚本语言，例如Java、JavaScript、ActiveX、VBScript或Perl等也要进行验证。

## 1.6 数据库测试

在Web应用技术中，数据库起着重要的作用，数据库为Web应用系统的管理、运行、查询和实现用户对数据存储的请求等提供空间。在Web应用中，最常用的数据库类型是关系型数据库，可以使用SQL对信息进行处理。在使用了数据库的Web应用系统中，一般情况下，可能发生两种错误，分别是数据一致性错误和输出错误。数据一致性错误主要是由于用户提交的表单信息不正确而造成的，而输出错误主要是由于网络速度或程序设计问题等引起的，针对这两种情况，可分别进行测试。

# 二、GUI基本测试内容

图形用户界面（ GUI ）对软件测试提出了有趣的挑战，因为 GUI 开发环境有可复用的构件，开发用户界面更加省时而且更加精确。同时， GUI 的复杂性也增加了，从而加大了设计和执行测试用例的难度。因为现在 GUI 设计和实现有了越来越多的类似，所以也就产生了一系列的测试标准。

## 2.1窗口

· 窗口是否基于相关的输入和菜单命令适当地打开？

· 窗口能否改变大小、移动和滚动？

· 窗口中的数据内容能否用鼠标、功能键、方向键和键盘访问？

· 当被覆盖并重新调用后，窗口能否正确地再生？

· 需要时能否使用所有窗口相关的功能？

· 所有窗口相关的功能是可操作的吗？

· 是否有相关的下拉式菜单、工具条、滚动条、对话框、按钮、图标和其他控制可为窗口使用，并适当地显示？

· 显示多个窗口时，窗口的名称是否被适当地表示？

· 如果使用多任务，是否所有的窗口被实时更新？

· 多次或不正确按鼠标是否会导致无法预料的副作用？

· 窗口的声音和颜色提示和窗口的操作顺序是否符合需求？

· 窗口是否正确地被关闭？

## 2.2下拉式菜单和鼠标操作

· 菜单条是否显示在合适的语境中？

· 应用程序的菜单条是否显示系统相关的特性（如时钟显示）？

· 下拉式操作能正确工作吗？

· 菜单、调色板和工具条是否工作正确？

· 是否适当地列出了所有的菜单功能和下拉式子功能？

· 是否可以通过鼠标访问所有的菜单功能？

· 文本字体、大小和格式是否正确？

· 是否能够用其他的文本命令激活每个菜单功能？

· 菜单功能是否随当前的窗口操作加亮或变灰？

· 菜单功能是否正确执行？

· 如果要求多次点击鼠标，是否能够在语境中正确识别？

### 3.3数据项

· 字母数字数据项是否能够正确回显，并输入到系统中？

· 图形模式的数据项（如滚动条）是否正常工作？

· 是否能够识别非法数据？

· 数据输入消息是否可理解？

# 三、性能测试

Apache JMeter工具

### 3.1简介

JMeter——一个100％的纯Java桌面应用，它是 Apache组织的开放源代码项目，它是功能和性能测试的工具。JMeter可以用于测试静态或者动态资源的性能（文件、Servlets、Perl脚本、Java对象、数据库和查询、ftp服务器或者其他资源）。

### 3.2测试环境：

硬件配置 软件环境

测试时间： 2015 年 12 月 25 日

服务器：Tomcat8.0

处理器：酷睿I5-2450M

内存：6G

数据库：MySQL

测试内容：接受用户参数、调用 javabean 、输出相关信息。

### 3.3测试内容：

（1）右键单击测试计划，选择添加-〉线程组

在线程组里设置负载信息，即线程属性。我向测试计划中增加相关负载设置是Jmeter需要模拟十个请求者，每个请求者在测试过程中并发请求，并且连续请求10次。

线程数： 10

Ramp-up period（inseconds）： 0

循环次数： 10

说明：线程数代表发送请求的用户数目，Ramp-up period（inseconds）代表每个请求发生的总时间间隔，单位是秒。假如我的请求数目是5，而这个参数是10，那么每个请求之间的间隔就是10／5，也就是2秒。如果设置为0就代表并发请求。Loop Count代表请求发生的重复次数，如果选择后面的forever（默认），那么 请求将一直继续，如果不选择forever，而在输入框中输入数字，那么请求将重复 指定的次数，如果输入0，那么请求将执行一次。

（2）添加HTTP默认请求值

右键单击线程组，选择添加-〉配置元件-〉HTTP默认请求值，以下是我设置的信息

名称： HTTP 默认请求值 该元素的名称

服务器名称或IP：

localhost 被测试服务器的 IP或者名字

端口号： 8080 服务器提供服务的端口号，我用的服务器是 Tomcat，所以端口号是8080

协议： http 发送测试请求时使用的协议

路径： http://123.103.9.193:9066/service 默认的起始位置。

（3）添加HTTP请求

右键单击线程组，选择添加-〉Sampler-〉HTTP请求，以下是我设置的信息

名称： HTTP 默认请求值 该元素的名称

服务器名称或IP：

localhost 被测试服务器的IP或者名字

端口号： 8080 服务器提供服务的端口号，我用的服务器是Tomcat，所以端口号是8080

协议： HTTP/IP

路径： http://123.103.9.193:9066/service默认的起始位置。

（4）添加监听器，存放测试结果。

右键单击测试计划，选择测试计划-〉添加-〉监听器-〉聚合报告

你也可以选择用表格查看结果、图形结果、查看结果树等。

### 3.4测试结果的分析说明

Label： 每个 JMeter 的 element （例如 HTTP Request ）都有一个 Name 属性，这里显示的就是 Name 属性的值

#Samples： 表示你这次测试中一共发出了多少个请求，我的测试计划模拟 10 个用户，每个用户迭代 10 次，因此这里显示 100

Average： 平均响应时间 —— 默认情况下是单个 Request 的平均响应时间，当使用了 Transaction Controller 时，也可以以 Transaction 为单位显示平均响应时间

Median： 中位数，也就是 50 ％用户的响应时间

90% Line： 90 ％用户的响应时间

Min： 最小响应时间

Max： 最大响应时间

Error%： 本次测试中出现错误的请求的数量 / 请求的总数

一般情况下，当用户能够在2秒以内得到响应时，会感觉系统的响应很快；当用户在2-5秒之间得到响应时，会感觉系统的响应速度还可以；当用户在5-10秒以内得到响应时，会感觉系统的响应速度很慢，但是还可以接受；而当用户在超过10秒后仍然无法得到响应时，会感觉系统糟透了，或者认为系统已经失去响应，而选择离开这个Web站点，或者发起第二次请求。故该系统的用户信息查询信息页面的在10到25人并发访问时，系统响应速度很快，25人到50人并发访问时速度还可以，50人到100人并发访问就比较慢了。