**编号**

JNDX_M

**本科生毕业设计（论文）**

**题目：** 面向业务的考试系统的设计与实现

计算机 学院 网络工程 专业

学 号 20122100014

学生姓名 叶豪

指导教师 王立彬 教授

二〇〇一六年六月

摘 要

论文摘要以浓缩的形式概括研究课题的内容，中文摘要在400字左右，外文摘要与中文内容相同，关键词一般以3～5个为妥，词与词之间以“；”为分隔.

设计总说明主要介绍设计任务来源、设计标准、设计原则及主要技术资料，中文字数要在1000～2000字以内，外文字数以500～1000个左右为宜，关键词一般以3～5个为妥，词与词之间以“；”分隔.

**关键词：**毕业论文；模板；规范；图；表

ABSTRACT

It is the English translation of the Chinese abstract. Font: Times New Roman, Word Size: 12. (same as “小四”).

**Keywords**: Thesis; template; criterion; figure; table

目 录

[第1章 绪论 1](#_Toc213724233)

[1.1二级标题 1](#_Toc213724234)

[1.1.1三级标题 1](#_Toc213724235)

[第2章 字体字号与页面设置 3](#_Toc213724236)

[2.1 字体字号 3](#_Toc213724237)

[2.1.1 标题写法 3](#_Toc213724238)

[2.2 页面设置 3](#_Toc213724239)

[第3章 图与公式的格式要求 5](#_Toc213724240)

[3.1 图 5](#_Toc213724241)

[3.1.1 图及图题标注 5](#_Toc213724242)

[3.1.2 公式及其标注 5](#_Toc213724243)

[第4章 表的格式要求 7](#_Toc213724244)

[4.1 表的格式 7](#_Toc213724245)

[4.1.1 表 7](#_Toc213724246)

[4.2 表的内容 7](#_Toc213724247)

[第5章 结论与展望 9](#_Toc213724248)

[5.1结论 9](#_Toc213724249)

[5.2不足之处及未来展望 9](#_Toc213724250)

[参考文献 11](#_Toc213724251)

[致 谢 12](#_Toc213724252)

[附录A： 作者在校期间发表的论文 13](#_Toc213724253)

[附录B： 14](#_Toc213724254)

第1章 绪论

随着信息科技发展，企业逐渐使用电子化的管理系统来进行自身业务的管理。企业对员工的培训中也着重考核员工对于其业务系统操作的熟练程度。本文论点主要围绕一个面向业务的考试系统的设计与实现。

1.1系统设计概要

本系统设计主要分为两部分：

（1）考试系统管理员的管理页面，其中包括

➀ 考生管理；

➁ 考试管理；

➂ 试题管理；

（2）普通考生的的页面，其中包括

➀ 考试页面；

➁ 考试成绩查询；

➂ 个人信息管理；

用户在登陆时选择用户类型（系统管理员或考生），登陆后会跳转到对应类型的页面。

1.2系统实现概要

**1.2.1 交互模式**

本系统采用 B/S 结构，用户通过浏览器与服务器进行交互。

**1.2.2 实现技术**

本系统使用JAVA作为服务器端开发语言，页面使用HTML/CSS/JS作为前端开发技术

开发环境：

JDK版本 1.7

Tomcat 7.1

MySQL 5.6

Eclipse Kepler

操作系统 Ubuntu 14.4

使用框架：

JAVA后台：Spring-MVC、Spring、Hibernate、JPA、FreeMarker

CSS、JS：BootStrap3、jQuery

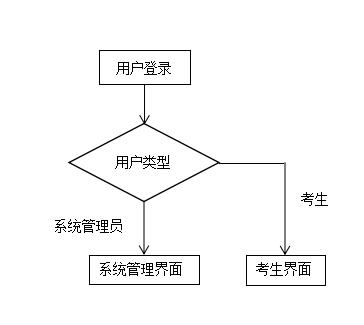
项目构建管理：

Maven、Git

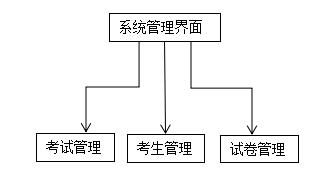
后台系统开发严格遵循MVC模式及表现层、业务层、持久层三层开发模式，使用Spring-MVC 作为表现层框架，Spring为业务逻辑层及Java Bean容器框架，Hibernate为持久层框架。页面 渲染使用FreeMarker，前端页面使用BootStrap及jQuery等开源框架。

1. 系统业务流程

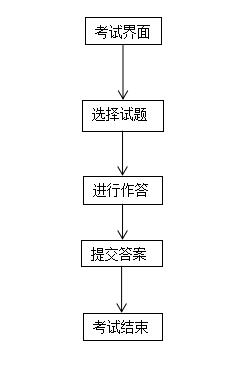
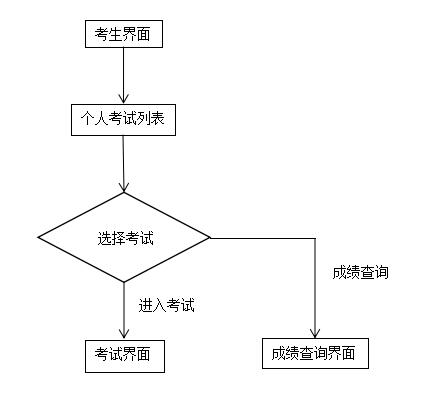
2.1 登陆流程



2.2 系统管理员业务流程



2.3 考生业务流程



第3章 考试系统域模型设计

3.1 数据库表设计

后台域模型共有4张实体域模型表，分别为用户表（User)，考试表(Exam)，试题表(ExamItem)，成绩表(ExamGrade)。

另外域模型还有2张关系表，分别是用户--考试关系表、考试--试题关系表。

**3.1.1 用户表设计**

用户表与实体类User对应，用于存储系统的用户信息，包括后台管理员与普通用户（考生），两者通过表中的user\_type字段进行区分。包含字段：

id 表中唯一主键

user\_type 用于区分用户类型

username 用户名，表中唯一

password 存储经过哈希函数哈希的用户登录口令

true\_name 用户真实姓名

no 用户编号

unit 用户单位

position 用户职位

basic\_info 用户备注信息

**3.1.2 考试表设计**

考试表与实体类(Exam)对应，用于存储系统考试信息，包括考试开始，结束时间。包含字段：

id 表中唯一主键

exam\_name 考试名字

generate\_time 考试生成时间

start\_time 考试开始时间

end\_time 考试结束时间

**3.1.2 试题表设计**

试题表与实体类(ExamItem)对应，用于存储试题信息。包含字段：

id 表中唯一主键

exam\_item\_name 试题名字

generate\_time 试题生成时间

question\_struct 试题内容，使用json格式存储，具体实现请参考后续章节

**3.1.3 成绩表设计**

成绩表与实体类(ExamGrade)对应，用于存储考生考试成绩信息。包含字段：

id 表中唯一主键

candidate\_answer 考生提交的对应考试成绩的答案

is\_submitted 用于检查考生考试提交状态

candidate\_id 指向考生表的外键

exam\_id 指向考试表的外键

**3.1.4 用户-考试关系表设计**

用户-考试关系表用来存储考生与考试之间的多对多关系，即一个考生可参加多个考试，一个考试也可以包含多个考生。包含字段：

exam\_id 指向考试表的外键

user\_id 指向用户表的外键

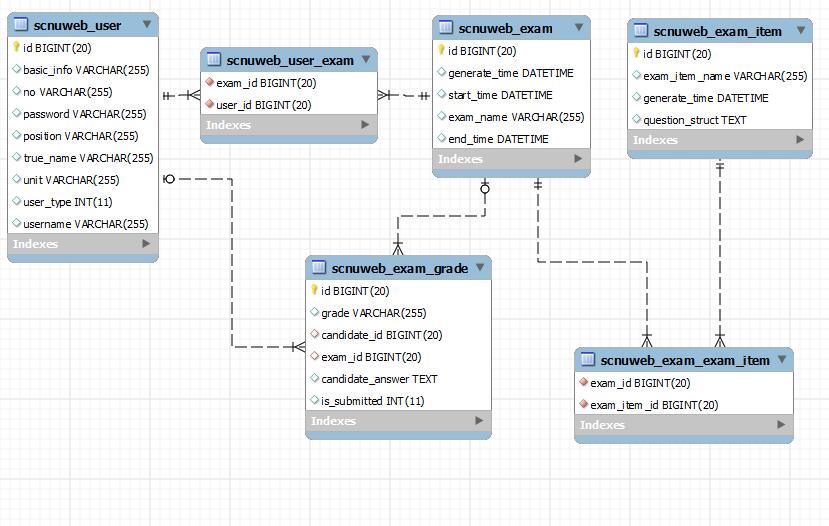
**3.1.5 考试-试题关系表设计**

考试-试题关系表用来存储考试与试题的多对多关系，即一场考试可以包含多个试题，一个试题也可被多个考试包含。包含字段：

exam\_id 指向考试表的外键

exam\_item\_id 指向试题表的外键

数据库EER图：



第4章 考试管理

后台管理员对考试管理主要是管理考试的起始时间，考试参与人员，考试试题。

4.1 考试时间管理

考试起始时间分别由2个DATE字段保存。管理员在考试管理界面可以选择起始时间。如果输入的起始时间不合法系统会拒绝接受。

起始时间用来控制考试的开始与结束时间。当考生选择进入考试与提交考试答案时系统会根据当前时间与起始时间的比较去判定操作是否有效。仅当当前时间在起始时间内系统才允许当前操作继续。

4.2 考试参与人员管理

考试与考生存在多对多的关系。即一场考试可有多名考生参加，一名考生可参加多场考试。所以在考试表与考生表（用户表）之间有一个张关系表，用来存储考试与考生之间的多对多关系。

管理员在考试管理页面可以选择已存在的考生进入当前考试。系统对于考试选定考生的实现采用ajax技术，在选择界面选择考生时只需勾选checkbox，即可完成选择。管理员在勾选时页面js脚本监听到后会把勾选的信息在页面不需要重新加载通过异步形式同步到后台。

4.3 考试试题管理

考试与试题存在多对多的关系。即一场考试可包含多个试题，一个试题也可被多个考试包含。考试表与试题表之间有一张关系表，用于存储考试与试题之间的多对多关系。

管理员在考试管理页面可以选择已存在的试题作为当前考试的题目。对于试题选定同样采用ajax技术，提高交互性。

第5章 考生管理

后台管理员拥有对考生的管理能力，即对考生进行添加，编辑，删除等操作。

5.1 考生列表

管理员在考生列表可以通过编号，用户名，姓名，单位来筛选得到考生列表。

考生列表每个单元行包含考生的基本信息，包括编号，用户名，姓名，单位，职务。管理员可通过每行的操作栏对考生进行编辑或删除操作。点击编辑即进入对应考生的编辑界面。

5.2 添加考生

管理员通过点击考生列表底端的“添加考生”按钮进入新增考生界面。在新增考生界面可以输入考生信息来完成考生的注册添加。在页面表单验证使用了jquery validator框架，通过js制定表单验证规则（正则表达式），当管理员输入的表单有误时及时提醒，增强页面交互性，体验性。然而对于像用户名这样全局唯一的字段必须保证其在提交到后台前未被其他考生使用。页面通过ajax把管理员输入的用户名异步传送到后台，后台检索并判断当前用户名是否可用，把反馈信息通过json回调到页面来响应用户。

5.3 编辑考生

管理员通过点击考生列表每行的操作栏中的编辑进入考生编辑界面。在考生编辑界面可以编辑考生的信息。编辑考生的页面表单同样使用了jquery validator来进行表单验证。管理员通过覆盖的方式更新考生信息。

5.4 删除考生

5.4.1 考生的级联删除

管理员通过点击考生列表每行操作栏中的删除按钮系统会再次确认是否删除，确认后系统将根据该考生id传至后台将对应考生纪录从数据库中删除。

由于考生表与考试表，考试成绩表均存在级联关系，故删除考生纪录时必须制定其关联表的级联规则。当删除考生时:

在考生-考试关系表中外键user\_id指向当前考生id的纪录都会被删除；

在成绩表中外键candidate\_id指向当前考生id的纪录都会被删除；

上面2种规则可通过JPA里面的注解设置，分别对应CascadeType.MERGE 和 CascadeType.REMOVE 。

5.4.2 考生的批量删除

考生列表每行的右侧有选定框，勾选多行后点击上方的删除按钮可实现批量删除。批量删除通过js脚本，用jquery迭代选择勾选行来生成删除考试的id列表并用ajax异步将删除列表提交到后台，后台解析列表后根据id迭代删除考生。

第6章 试题管理

管理员组织一场新的考试时，除了需要选定参加考试的考生，还要选定考试内容，即试题。一场考试可包含多个试题。管理员在组织新考试的界面可以选择多个试题作为考试的试题内容。

6.1 添加试题

考试试题

第5章 结论与展望

5.1结论

XXX

5.2不足之处及未来展望

XXX

参考文献

1. 张卫平. 开关变换器的建模与控制[M]. 北京：中国电力出版社，2006，15-88.
2. 曹文思，杨育霞. 基于状态空间平均法的BOOST变换器仿真分析[J].系统仿真学报， 2007，19(6)：1329- 1334.
3. 伍言真. DC/DC开关变换器建模分析及其变结构控制方法的研究[D]. 广州：华南理工大学，1998.
4. Takagi T, Sugeno M. Fuzzy identification of systems and its applications to modeling and control [J]. IEEE Trans on Systems, Man and Cybernetics, 1985, 15(2): 116-132.
5. Sugeno M, Kang G T. Fuzzy modeling and control of multilayer incinerator [J]. Fuzzy Sets Syst., 1986, 18(2): 329-346.
6. 张勇. [HOPE-Ⅰ模拟器的实现](http://epub.cnki.net/grid2008/Detail.aspx?dbname=CPFD2004&filename=ZGWW200410001063&filetitle=HOPE-%e2%85%a0%e6%a8%a1%e6%8b%9f%e5%99%a8%e7%9a%84%e8%ae%be%e8%ae%a1%e4%b8%8e%e5%ae%9e%e7%8e%b0)[C]，第三届中国学术会议论文集[C]．北京：北京邮电大学出版社, 2001：449-740.

致 谢

致谢应以简短的文字对在课题研究和设计说明书（论文）撰写过程中曾直接给予帮助的人员或单位表示自己的谢意，这不仅是一种礼貌，也是对他人劳动的尊重，是治学者应有的思想作风，比如：本文是在导师XXX教授和XXX讲师的悉心指导下完成的，表示谢意!

感谢XX.

附录A： 作者在校期间发表的论文

[1]作者．文献题名[J]．刊名，出版年份，卷号(期号) ：起止页码．

附录B： XX

附录是对于一些不宜放在正文中，但有参考价值的内容，可编入毕业设计（论文）的附录中，例如公式的推演、编写的程序等；如果文章中引用的符号较多时，便于读者查阅，可以编写一个符号说明，注明符号代表的意义.一般附录的篇幅不宜过大.