

大连理工大学

硕士学位论文

基于遗传算法的考试系统的设计与实现

姓名：孟禹

申请学位级别：硕士

专业：软件工程

指导教师：李明楚

20081109

摘 要

随着计算机网络技术在教育领域的广泛应用，传统的考试方式正面临着重大的变革，而利用网络来进行考试则是一种先进的考试方式，它有很多传统考试所不具有的优越性。它不但可以节省大量的人力、物力，还可以提高考试的效率和质量。

本文构建了一个基于 B/S 多层结构的、以 JSP 技术和 SQL 后台数据库为基础的在线考试系统。同时，论文立足于面向对象的可视化建模，应用 UML 统一建模语言对网络考试系统的结构、功能等要素进行了可视化描述，并为系统开发的各个阶段建立了可视化模型。本文还对考试系统中的自动组卷算法进行了详细的描述。

网络考试系统主要具备以下功能：考生信息管理、试题库管理、在线考试、计算机自动阅卷等功能。具有六个特点：一是实现题库数字化。将试题输入计算机，利用数据库技术对试题库进行管理，便于实现试题的保存、分类、检索和制作试卷。二是随机自动生成试卷。按照教师制定的组卷策略随机生成试卷，提高了出卷效率和质量。同时避免考前的试题泄密，确保考试公正合理。三是题库适时更新。教师可以根据对考试结果的评估和分析，或者根据考试对象，对试题库进行增加、删除、修改、更新，增强了题库的生命力。四是有效保存考试数据。考生的试卷、考试成绩都存储于数据库中，方便评阅试卷，提高判卷的准确度，使考试过程规范化。五是加强安全性控制。题库的编辑、修改，试卷的查询、考试成绩的查询，都需要进行身份验证。六是使用方便快捷。计算机考试系统提供了方便的考试环境，既节省了印刷试卷的成本，又减轻了教师批卷的负担。

关键词：考试系统；组卷；自动阅卷

The Design and Implementation of Examination System Based on Web

Abstract

With the wider applications of the network technology in the field of education, the traditional way of examinations is facing major changes, and the use of the Internet to conduct examination is an advanced way of examination, it has a lot of advantages over the traditional examination. It can not only save a lot of manpower, material, but also to improve the efficiency and quality of the examination.

In this paper, the article discussed the method of the realizing of the online examination system which is a B/S multi-layer structured and based on the JSP and SQL database technology. The paper applies UML to construct model from the system structure, process and function, describing the process from modeling, design modeling to implement modeling. And the paper describes the algorithm of auto-generating test paper in detail.

Network test systems have the following functions: information management candidates, the test library management, online examinations, the computer automatically scoring functions. Have six characteristics: First, the realization of the number of questions. Questions will be entered into the computer, using database technology to the test library management, to facilitate the realization of the questions preservation, classification, retrieval and production of papers. The second is automatically generated random papers. Teachers in accordance with the development of the test paper randomly generated strategy papers, the volume increased efficiency and quality. At the same time to avoid a test of the leak test to ensure a fair and rational examination. Third, the bank updated in a timely manner. Teachers can be based on the results of the examination and assessment of the analysis or examination in accordance with the target, the test for the Treasury to add, delete, modify, update and enhance the vitality of the bank. The fourth is to preserve an effective test data. Candidates of the papers, examination results are stored in the database to facilitate the marking examination papers, marking to improve the accuracy, so that the process of standardized tests. The fifth is to strengthen the security control. The bank editing, revision, the paper's inquiry, the examination results of the inquiry will require authentication. Six is convenient. Examination of the computer system to facilitate the examination of the environment, not only saves the cost of printing papers, but also to reduce the burden on teacher.

Key Words: Examination System; Generating Test Paper; Auto-marking Test Paper

大连理工大学学位论文独创性声明

作者郑重声明：所呈交的学位论文，是本人在导师的指导下进行研究工作所取得的成果。尽我所知，除文中已经注明引用内容和致谢的地方外，本论文不包含其他个人或集体已经发表的研究成果，也不包含其他已申请学位或其他用途使用过的成果。与我一同工作的同志对本研究所做的贡献均已在论文中做了明确的说明并表示了谢意。

若有不实之处，本人愿意承担相关法律责任。

学位论文题目：基于遗传算法的考试系统的设计与实现

作者签名：孙平 日期：2008 年 11 月 10 日

大连理工大学学位论文版权使用授权书

本人完全了解学校有关学位论文知识产权的规定，在校攻读学位期间论文工作的知识产权属于大连理工大学，允许论文被查阅和借阅。学校有权保留论文并向国家有关部门或机构送交论文的复印件和电子版，可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印、或扫描等复制手段保存和汇编本学位论文。

学位论文题目：基于遗传算法的考试系统的设计与实现

作者签名：高禹

日期：2008年11月10日

导师签名：李明庆

日期：2008年11月10日

1 绪论

1.1 论文的选题背景和研究意义

考试一直是学校教学活动中最重要的环节之一，其主要的宗旨在于检测知识水平。传统的考试都是采用纸、笔为介质的手工考试方式^[1]，即使在目前，这种传统的考试方式仍然广泛使用。但是，随着教学手段的不断发展，考试的类型和要求也随之不断提高，每逢考试，学校花费大量的人力和物力去组织考试。而在考试结束后，对大量考生的考试成绩进行统计分析，也是一件十分烦琐并且容易出错的事情。还有，采用传统的考试方式，试卷的反馈能力弱，试题的选择也具有较大的随意性，试卷的难度、区分度等指标也难以估算。成绩分析也仅仅停留在平均分、及格率等表面层次上，更难以统计出学生对各个知识点的掌握程度并以此发现教师在教学中的薄弱环节。而且，即便统计出成绩之后，学生为查看成绩给教师和教务部门找的麻烦也不少，因此传统的考试方式已经不能适应现代考试的需要。

随着计算机技术和网络技术的发展 and 普及，特别是 Internet 的迅速发展和广泛应用，将分布在世界各地的各种计算机系统及网络用户连接在一起，使人们可以方便地通过共同的网络通信协议，在不同的网络和操作系统间交换数据。近年来，现代远程教育作为现代教育技术的形式，给传统的教育思想和技术带来了革命性的变革，已成为建设学习型社会，推动教育信息化进程的必然要求^[2]。它可以最大限度地利用各种资源，将各个学校的优势教育资源、优秀的教师、最好的教学成果通过网络传播到四面八方，使任何人、在任何时间、任何地点、从任何章节开始、学习任何课程，为学生的自主学习、个性化学习提供了必要条件，同时也需要提供相应的测试或考试系统以检测学习的成效。这些都使得考试的手段和媒介发生着革命性的变化，传统的考试方式和手段正面临着强烈的冲击。在网络技术已经逐渐渗入社会生活各个层面的今天，传统的考试方式也面临着变革，而网络考试则是一个很重要的方向。基于 Web 技术的网络考试系统是一种先进的考试方式，它比传统的考试方式更具优越性：

(1) 考试形式灵活多样。借助因特网进行考试，既可以在本地进行，也可以在异地进行，大大拓展了考试的灵活性。同时考试不受时间和空间的限制。

(2) 节约考试成本。基于 Web 技术的在线式考试系统实现无纸化，不需要印刷试卷，可以节省大量的人力、物力和财力。

(3) 数字化的试题库。在建立试题库时，将课程考试的题目按要求录入到数据库中，既利于长期保存，减少老师重复翻查资料进行出题的工作；又能够试题库进行科学的管

理，便于对考试内容进行分类、检索，为编制试卷提供了高效率 and 低成本的手段；教师还可以依据考试的分析结果，及时对试题库进行增加、删除、修改、更新，增强了题库的生命力。

(4) 考试系统安全高效。试题内容、考试答案存放在服务器中，考生需通过身份验证才能够使用考试系统，这样考试的公平性、答案的安全性可以得到有效的保证。同时题库的编辑、修改，试卷的查询、考试成绩的查询，都需要进行身份验证。

(5) 自动判卷快捷准确。考试系统采用大量标准化试题，便于计算机根据评分标准自动评分，无需人工阅卷，减轻了教师改卷的负担，提高了阅卷的效率和准确度^[3]，避免了人工判卷时经常出现的错判、漏判和分数计算错误等情况，使考试过程更加规范化。还可以直接把成绩送到数据库中，进行统计、排序等操作。

与此同时，由于办学规模的扩大，目前普通高校普遍存在师资力量不足的现象，教师工作量大，将大部分精力放在上课、作业、考试上，难以抽出更多的时间提高自身素质或进行教改和科研工作。同时，作为新形势下教学方法和考试方法的数字化改革，改变传统的计算机基础课程的考试手段具有重要意义^[4]。

1.2 论文的主要内容

本论文的主要内容是远程考试系统的研究与实现。采用了面向对象分析与设计的思想，使用 UML 统一建模语言对远程考试系统建模，研究和开发了适用于本项目特点的基于 B/S 模式的网络考试系统。论文共分六章，阐述了该系统的总体结构，功能特点，设计方案。并对该系统所采用的相关技术和实现方法进行了重点介绍。

第一章论述了考试系统的开发背景及现实意义。

第二章简要介绍了系统开发过程中所用到的各项技术，如 JSP、UML 方面的相关知识。

第三章为考试系统的需求分析，分析了考试系统的功能需求，并对系统进行了用例建模。

第四章对考试系统中的各主要功能及数据库进行了详细设计。

第五章详细介绍了系统中的一项关键技术，即基于遗传算法的自动组卷算法。

第六章是在线考试系统的实现，并以系统中的几项主要功能为例，详细介绍了其工作原理及工作流程。

2 系统开发中的相关技术介绍

2.1 统一建模语言 (UML)

软件系统的开发是一个非常复杂的系统过程,为了更好的理解整个软件系统,需要建立系统模型。统一建模语言 (Unified Modeling Language, UML)^[5-6]是一种用来绘制软件蓝图的标准语言。通过 UML 可以对软件制品进行可视化、详述、构造和文档化。从企业信息系统到基于 Web 的分布式应用,甚至对于严格的实时嵌入式系统,都可以通过 UML 进行建模。这种建模语言得到了“UML 伙伴联盟”的应用与反馈,并得到工业界的广泛支持,由 OMG 组织 (Object Management Group) 采纳作为业界标准^[7]。目前,OMG 组织已经将 UML 作为公共可得到的规格说明提交给国际标准化组织 (International Organization for Standardization, ISO) 进行国际标准化,UML 即将成为信息技术的国际标准^[8]。

2.1.1 UML 的概念模型

为了理解 UML,需要形成一个语言的概念模型。UML 包括三个主要要素:UML 的基本构建块、支配这些构建块放在一起的规则和一些运用于整个 UML 的公共机制。这三部分构成了 UML 的概念模型^[9]。

(1) UML 的基本构建块

UML 的词汇表中包括三种构建块:事物、关系和图。事物是模型中最有代表性成分的抽象,关系把事物结合在一起,图聚集相关的事物。

UML 中事物可分为四种:结构事物、行为事物、分组事物、注释事物。

结构事物:结构事物通常是 UML 模型中的静态部分,描述概念或物理元素。主要包括以下七种:类、接口、协作、用例、主动类、构件和节点。类是对一组具有相同属性、相同操作、相同关系和相同语义的对象的描述。一个类实现一个或多个接口。接口描述一个类或构件的一个服务的操作集。协作定义一个交互,它是由一组共同工作以提供某协作行为的角色和其他元素构成的一个群体,这些协作行为大于所有元素的各自行为的总和。用例是对一组动作序列的描述。主动类是一个类,它的对象至少拥有一个线程或进程。构件是系统中物理的、可替代的部件,它遵循且提供一组接口的实现。节点是在运行时存在的物理元素,它表示一种可计算的资源,至少有一些记忆能力和处理能力。

行为事物：行为事物是 UML 模型的动态部分，描述跨时间和空间的行为，主要包括交互和状态机。交互描述的是在特定语境中共同完成一定任务的一组对象之间消息的交换。状态机描述一个对象或一个交互在生命期内响应事件所经历的状态序列。

分组事物：分组事物是 UML 模型的组织部分。在所有的分组事物中，最主要的分组事物是包。包是把元素组织成组的一种机制。结构事物、行为事物甚至其他的分组事物都可以组合成包。

注释事物：注释事物是 UML 模型的解释部分。这些注释事物用来描述、说明和标注模型的任何元素。

在 UML 中主要有四种关系：依赖、关联、泛化和实现。依赖是两个事物间的语义关系，表示其中一个事物（独立事物）发生变化会影响另一个事物（依赖事物）。关联是一种结构关系，它描述一组链，即对象之间的连接。聚合描述了整体和部分之间的结构关系。泛化是一种特殊/一般关系，特殊（子元素）的对象可替代一般元素（父元素）的对象。实现是指一个类定义一个操作接口，而另一个类实现这个操作接口的这样一种关系。

在 UML 中包括九种图：类图、对象图、用例图、顺序图、协作图、状态图、活动图、构件图和实施图^[10-11]。

类图（Class Diagram）展现一组对象、接口、协作和它们之间的关系，是 UML 建模中最常见的图。

对象图（Object Diagram）展现一组对象以及它们之间的关系，用以说明类图中所反应的事物实例的数据结构和静态快照。对象图表达系统的静态设计视图或静态过程视图，除了现实和原型方面的因素外，它与类图作用是相同的。

用例图（Use Case Diagram）展现一组用例、参与者（一种特殊的类）及其之间的关系。可以用用例图描述系统的静态使用情况。在对系统行为组织和建模方面，用例图是相当重要的。

顺序图（Sequence Diagram）和**协作图（Collaboration Diagram）**统称为**交互图（Interaction Diagram）**，交互图展现了一种交互，它由一组对象及其之间的关系组成，包括在它们之间可能发送的信息。其中，顺序图（Sequence Diagram）是一种强调消息的时间顺序的交互图，而协作图（Collaboration Diagram）强调收发信息的对象的结构组织。顺序图和防作图是同构的，这意味着它们是可以相互转换的。

状态图（State Diagram）展现一个状态机，它由状态、转换、事件和活动组成。

活动图 (Activity Diagram) 显示了系统中从活动到活动的流, 它对于系统的功能建模特别重要, 强调对象间的控制流程。

组件图 (Component Diagram) 展现了一组构件之间的组织和依赖关系。

配置图 (Deployment Diagram) 展现系统运行时处理节点以及其中的构件的配置。

需要说明的是 UML 并不限定仅使用以上 9 种图, 开发工具可以提供符合 UML 语义的其他种类的图, 但到目前为止, 这 9 种图在实际应用中是最常用的。使用这 9 种图能基本实现应用系统的建模问题。

(2) UML 规则

UML 用于描述事物的语义规则分别是: 为事物、关系和图命名; 给一个名字以特定含义的语境, 即范围; 怎样使用或看见名字, 即可见性; 事物如何正确、一致地相互联系, 既完整性; 运行或模拟动态模型的含义是什么, 即执行。另外, UML 还允许在一定的阶段隐藏模型的某些元素、遗漏某些元素以及保证模型的完整性, 但模型逐步地要达到完整和一致。在软件开发生命周期内, 随着系统细节的展开和变动, 不可避免地要出现一些不太规范的模型。UML 的规则鼓励你专注于最重要的分析、设计和实现问题。这些问题将促使模型随时间的推移而具有良好的结构。

(3) UML 的公共机制

有四种在整个 UML 语言中一致应用的机制, 使该语言变得较为简单。这四种机制是规格说明、修饰、通用划分和扩展机制。

2.1.2 UML 的建模机制

从应用的角度看, 当采用面向对象技术设计系统时, 首先是描述; 其次根据需求建立系统的静态模型, 以构造系统的结构; 第三步是描述系统的行为, 其中在第一步与第二步中所建立的模型都是静态的, 包括用例图、类图、对象图、构件图和实施图等 5 个图形, 是标准建模语言 UML 的静态建模机制。其中第三步中所建立的模型或者可以执行, 或者表示执行时的时序状态或交互关系。它包括状态图、活动图、顺序图和合作图等四个图形, 是标准建模语言 UML 的动态建模机制。因此, 标准建模语言 UML 的主要内容也可以归纳为静态建模机制和动态建模机制两大类^[12]。

(1) 静态建模机制

任何建模语言都以静态建模机制为基础, 标准建模语言 UML 也不例外。UML 的静态建模机制包括用例图 (Use Case Diagram)、类图 (Class Diagram)、对象图 (Object Diagram)、包 (Package)、构件图 (Component Diagram) 和实施图 (Deployment Diagram)。

① 用例图：用例图展现了一组用例、参与者以及它们之间的关系。可以用用例图描述系统的静态使用情况。在对系统行为组织和建模方面，用例图是相当重要的。

② 类图：类图展现了一组类、接口和协作及它们间的关系，在建模中所建立的最常见的图就是类图。用类图说明系统的静态设计视图，包含主动类的类图—专注于系统的静态进程视图。系统可有多个类图，单个类图仅表达了系统的一个方面。要在高层给出类的主要职责，在低层给出类的属性和操作。

③ 对象图：对象图展示了一组对象及它们间的关系，用对象图说明类图中所反映的事物实例的数据结构和静态快照。对象图表达了系统的静态设计视图或静态过程视图，除了现实和原型方面的因素外，它与类图的作用是相同的。

④ 构件图：构件图展现了一组构件之间的组织和依赖，用于对源代码、可执行的发布、物理数据库的系统建模。

⑤ 实施图：实施图展现了对运行时处理节点以及其中构件的配置。它描述系统硬件的物理拓扑结构（包括网络布局和构件在网络上的位置），以及在此结构上执行的软件（即运行时软件在节点中的分布情况）。用实施图说明系统结构的静态部署图，即说明分布、交付和安装的物理系统。

(2) 动态建模机制

系统的动态行为主要是由交互图（顺序图和协作图）、状态图和活动图来表达。

① 顺序图：顺序图展现了一组对象和由这组对象收发的消息，用于按时间顺序对控制流建模。用顺序图说明系统的动态视图。

② 协作图：协作图展现了一组对象，这组对象间的连接以及这组对象收发的消息。它强调收发消息的对象的结构组织，按组织结构对控制流建模。

③ 状态图：状态图展现了一个特定对象的所有可能状态以及由于各种事件的发生而引起的状态间的转移。一个状态图描述了一个状态机，用状态图说明系统的动态视图。它对于接口、类或协作的行为建模尤为重要，可用它描述用例实例的生命周期。

④ 活动图：活动图是一种特殊的状态图，描述需要做的活动、执行这些活动的顺序（多为并行的）以及 workflow（完成工作所需要的步骤）。它对于系统的功能建模特别重要，强调对象间的控制流程。高层活动图用于表示需要完成的一些任务，即用于分析用例，理解涉及多个用例的 workflow、多线程及并行，显示相互联系的行为整体，还可用于对企业过程建模，对系统的功能建模。低层活动图用于表示类的方法。但活动图不适用于描述动作与对象间的关系，显示对象间的合作以及显示对象在生命周期内的运转情况。

2.2 JSP 概述

当今世界, 计算机技术飞速发展, 网络正深入到人们生活的每一个部分, 人们从网络上获取信息、享受服务、交流情感, 网络正在迅速改变着我们的生存方式。同时, 网络的发展也给了 Web 技术提供了广阔的应用前景, 各种基于 Web 的技术正越来越受到人们的欢迎。

JSP 是 Java Server Page 的简称, 是 Sun 公司推出的服务器端嵌入式脚本语言。JSP 是 J2EE 平台面向多层体系结构下表示层模式的关键技术。JSP 技术基于平台和服务器的互相独立, 输入支持来自广泛的, 专门的, 各种工具包, 服务器的组件和数据库产品开发商所提供。

由于 JSP 页面的内置脚本语言是基于 Java 程序设计语言的, 而且所有的 JSP 页面都被编译成为 Java Servlet, JSP 页面就具有 Java 技术的所有优点, 包括健壮的存储管理和安全性。

几乎所有平台都支持 Java, JSP+JavaBean 可以在所有平台下通行无阻。著名的 Web 服务器 Apache 已经能够支持 JSP。由于 Apache 广泛应用在 NT、Unix 和 Linux 上, 因此 JSP 有更广泛的运行平台。虽然现在 NT 操作系统占了很大的市场份额, 但是在服务器方面 Unix 的优势仍然很大, 而新崛起的 Linux 更是来势不小。从一个平台移植到另外一个平台, JSP 和 JavaBean 甚至不用重新编译, 因为 Java 字节码都是标准的与平台无关的。

Java 中连接数据库的技术是 JDBC (Java Database Connectivity)。很多数据库系统带有 JDBC 驱动程序, Java 程序就通过 JDBC 驱动程序与数据库相连, 执行查询、提取数据等操作。Sun 公司还开发了 JDBC-ODBC bridge, 用此技术 Java 程序就可以访问带有 ODBC 驱动程序的数据库, 目前大多数数据库系统都带有 ODBC 驱动程序, 所以 Java 程序能访问诸如 Oracle、Sybase、SQL Server 和 Access 等数据库^[13]。

使用 JSP 技术, Web 页面开发人员可以使用 HTML 或者 XML 标识来设计和格式化最终页面。使用 JSP 标识或者小脚本来生成页面上的动态内容。生成内容的逻辑被封装在标识 JavaBeans 组件中, 并且捆绑在小脚本中, 所有的脚本在服务器端运行。如果核心逻辑被封装在标识和 Bean 中, 那么其他人, 如网站管理员和页面设计者, 能够编辑和使用 JSP 页面, 而不影响内容生成。在服务端, JSP 引擎解释 JSP 标识和小脚本。生成所请求的内容 (例如, 通过访问 JavaBean 组件, 使用 JDBC 技术访问数据库或者包含文件), 并且将结果以 HTML (或者 XML) 页面的形式发送回浏览器。这有助于作者保护自己的代码, 而又保证任何基于 HTML 的 Web 浏览器的完全可用性^[14-15]。

目前, JSP 与 ASP、PHP 技术都是流行的应用于服务器端脚本技术, 其中, JSP 能用 JavaBeans 技术封装, 达到商业逻辑的目的, ASP 与之对应的是采用了 COM 组件, 因此两者都能用于开发商业逻辑, PHP 则更像一个纯脚本, 仅适合于普通的网站开发, 而不能用作商业逻辑。JSP 与 ASP 相比, JSP 具有跨平台、安全、强大的可伸缩性的特点, 虽然 ASP 的最新版本 ASP.NET 与原来的 ASP 性质完全是不同的, 它和 JSP 一样, 都是一种编译性的脚本, 本身完全与 .Net 框架结合, 完美的支持 XML 的各种应用。但是无论是 ASP 还是 ASP.NET, 它都摆脱不了 Windows 平台本身, 这是 ASP 一个致命的缺点^[16]。

2.3 Servlet 技术

Servlet 作为 Java 技术平台支持 Web 应用的突破口, 解决了 CGI 和专用 API 存在的问题, ServletAPI 很简单, 几乎所有的 Web 服务器都支持 Servlet 程序。它通过多线程解决了 CGI 的性能问题。同时, Servlet 是用 Java 编写的, 继承了 Java 的跨平台特性, 以及兼容性和安全性, 因此 Servlet 被认为 CGI 的替代语言。

Servlet 是 Java2.0 中新增的功能, 它与 Applet 相对应, Applet 运行在客户端的浏览器上, 而 Servlet 运行在服务器端。Servlet 是服务器小程序, 运行于服务器端, 与运行于客户端的小程序 Applet 相对应。它们都是由 Java 编写的小程序, 都没有 main 方法, 只有一些特定的方法用于启动、执行和退出, 不同的是 Servlet 不提供用户界面。

(1) Servlet 运行机理:

① Servlet 是一个由 Servlet 容器管理的可以产生动态内容的 Web 构件;

② Servlet 是一个小的、与平台无关的 Java 类, 它实现了 Java 的 Servlet 接口, 被编译成与平台无关的字节码, 然后加载到 Servlet 容器中运行, 而且整个生命周期都由 Servlet 容器管理。

Servlet 同客户端的交互遵循“请求/响应”模型, 这个模型基于超文本传输协议 (HTTP) 的基本机理, 所以 Servlet 很适合开发 Web 应用。这些请求/响应过程也由 Servlet 容器管理, Servlet 容器是运行在 Java 虚拟机上的管理器, 它同 Web 服务器或应用服务器相结合提供网络服务, 在 Web 应用中, Servlet 可用于接收来自 Web 浏览器的 HTTP 请求, 动态的生成响应, 然后发送包含处理结果的 HTTP 响应到浏览器^[17]。

(2) Servlet 的工作流程为:

① 客户端 Web 浏览器通过 HTTP 协议提出请求;

② Web 服务器接受该客户端请求并将其发送给 Servlet, 如果需要处理的 Servlet 尚未被加载, Web 服务器加载这个 Servlet 给 Java 虚拟机并执行它;

③ Servlet 接收该 HTTP 请求并进行相应处理;

④ Servlet 向 Web 服务器返回应答;

⑤ Web 服务器把从 Servlet 收到的应答发送给客户端浏览器。

Servlet 程序是在 Servlet 容器管理下运行, Servlet 容器按照一个严格的生命周期管理 Servlet, 这个生命周期定义了 Servlet 如何被加载, 实例化, 初始化, 处理客户端请求, 最后结束服务, 这个周期称为 Servlet 生命周期。生命周期的每一步对应了 Servlet 接口的不同方法, Servlet 程序实现这些方法, Servlet 容器则按照生命周期调用这些方法, 管理和运行 Servlet 程序的运行的实例。

(3) Servlet 的优势:

① Servlet 是持久的, Servlet 只需 Web 服务器加载一次, 而且可以在不同的请求之间保持服务。

② Servlet 是快速的。Servlet 在响应请求后, 能够在后台持续运行, 在多线程中处理多个请求。

③ Servlet 是平台无关的。Servlet 是使用 Java 编写的, 所以是与平台无关的。

④ Servlet 能够保持状态。Servlet 一旦被服务器加载, 就能够保持其状态信息, 只要 Servlet 就没有被改变, 它就不需要被重新加载。

(4) JSP 与 Servlet 的比较^[18-19]:

JSP 与 Servlet 是类似的。JSP 的运作方式取决与 JSP 引擎的运作方式, 它是编译式运作方式。JSP 引擎的任务主要是把工作移交给 Servlet 引擎。

除去编译的部分, JSP 和 Servlet 的差异不大, JSP 引擎是架构在 Servlet 引擎之上的, 以 Servlet 的形式存在, 利用 Servlet 引擎将 JSP 文档转译成 Servlet 原始码, 再调用 Java 编译器编译成 Servlet, 所以第一次执行 JSP 需要较多的时间, 以后执行 JSP 与 Servlet 所需的时间相同。

SUN 公司先推出 Servlet, 其功能较强大, 体系设计也先进, 但在输出 HTML 语句时还是采用了 CGI 方式, 是一句一句的输出, 所以编写和修改 HTML 非常不方便。一个网络项目一般分成表示层, 业务层和数据层。Servlet 用来写业务层很强大, 但对于写表示层就很不方便。JSP 主要是为了方便写表示层而设计的。它既可以书写表示层, 也可以写事务层。一般 JSP 书写表示层的内容, 而把数据实现两层的结合。Servlet 执行效率比 JSP 高, 但 JSP 编写方便, 因此, 在需要执行效率高的地方, Servlet 仍然有它的优势。

2.4 JavaBean

2.4.1 JavaBean 概述

Java 的软件称为 JavaBean, 或简称 Bean。按照 JavaSoft 给出的定义, Bean 是能够在构造工具中进行可视化操作的可重用软件。由于 Bean 是用 Java 写的, 所以 Java 环境/语言所固有的特性将会在 Bean 中得到很好的体现, 从而可使 Bean 极少拥有其他语言经常出现的内存泄露问题及指针错用问题, 并能够实现不同硬件平台之间的互操作特性 (JavaBean 通过其内部的 Java 虚拟机实现对多平台的支持)。

Bean 是一个可以在 Java 编程环境中可视化处理的可重用软件组件。一个 Bean 和一个 JavaApplet 相似, 是一个非常简单的遵守某些协议的 Java 类。Bean 通常含有源文件, 可以包装成一个 JAR 格式的文件。开发者可以通过支持 Bean 的开发工具, 直接使用现在的 Bean, 也可以在开发工具容器中, 对组件进行必要的修改和测试, 而不用编写和编译程序。在 Java 模型中, 组件可以修改或与其他组件组合生成新组件或完整的应用程序^[20]。

大多数 Bean 通常是中小型控制程序, 但也可以是生成包装整个应用程序的 Bean。Bean 可以表示为简单的 GUI, 如按钮, 列表等。但也可以生成不可见 Bean, 用于接收事件和在幕后工作。所以, 可把 Bean 看成是用于接收事件和处理事件以便进行某个操作的组件。一个 Bean 由 3 部分组成^[21]:

(1) 属性 (Property): Bean 属性就是对象的属性, 提供了属性读取和设置的接口支持。

(2) 方法 (Method): Bean 本身就是 Java 对象, 它严格遵守面向对象的类设计逻辑, 不让外界访问其任何字段, 方法调用是调用 Bean 的唯一途径。

(3) 事件 (Event): Bean 与其他组件交流信息的主要方式是发送和接收事件。这与对象之间信息通信类似。

2.4.2 JavaBean 的用途

JavaBean 在 Web 应用程序中的用途主要有以下方面:

(1) 用来封装公共的功能。

(2) 当某些功能在程序中被多次使用, 就可以 JavaBean 将它们封装成一个可重用的构件, 这既方便了程序设计, 也使程序结构更加合理。

(3) 用来实现业务逻辑的实体。

(4) 一个应用系统通常分成表示逻辑和业务逻辑等部分。表示逻辑负责实现应用系统的页面显示部分，这部分用 HTML, XML, JSP, Servlet 等技术来实现。业务逻辑负责实现应用程序的业务功能的部分，如取送数据到数据库等，这部分可用成熟的面向对象的技术和语言来实现，更好的方法是使用 JavaBean 来实现业务逻辑，因为 JavaBean 可封装业务实体的功能，达到软件重用的思想，简化了程序设计。

(5) 作为在程序各个部分之间传递数据的实体。可以在程序的业务逻辑中设置 JavaBean 的属性，在其他程序中读取 Bean 的属性而获得数据。这样，JavaBean 作为程序各个部分之间传递数据的实体，达到了程序中传递数据的目的。

2.5 JDBC

数据库应用结构采用三层结构。客户端浏览器不直接访问位于服务器上的后台数据库，而是通过位于中间层的 Web 服务器和支持 Web 的应用服务器访问数据库。中间层程序可以使用 Servlet, JSP, JavaBean 来实现，中间层通过 JDBC 访问数据库^[22]。有了 JDBC 就可以让程序员用 Java 语言来编写完整的数据库方面的应用程序。另外也可以对保存在多种不同的数据库管理系统中的数据进行操作，而与数据库管理系统中数据存储格式无关。同时由于 Java 语言与平台的无关性，所以不必在不同的系统平台下编写不同的数据库应用程序^[23]。

2.5.1 JDBC 的使用方法

数据库系统在现代信息系统中已经成为一个不可或缺的组成部分，现在很难想象一个信息系统不建立在数据库的基础之上。在使用 Java 技术开发 Web 应用程序时，对数据库的访问技术是最重要的组成技术之一。

JDBC 是一种用于执行 SQL 语句的 Java API, 它由一组用 Java 编程语言编写的类和接口组成，它为数据库开发人员提供了一组标准的 API。它同 Java 中的其他 API 一样，可以用于 JSP 程序中构造各种数据库对象，编写数据库应用程序，与数据库建立连接，发送 SQL 语句和处理结果。ODBC 是一个 Windows 平台上的访问数据库的标准 API，使用 ODBC 使得应用程序同具体的数据库接口分离开来，应用程序对不同数据库具有兼容性和可移植性，无论任何一个数据库，采用 ODBC 接口都可以按同样的方式进行访问。JDBC 提供给 Java 程序使用的，它将各种数据库的差异对 Java 程序屏蔽了起来，Java 程序可以使用同样的可移植的接口访问数据库，是 Java 的应用程序屏蔽了数据库领域，同时保持了 Java 语言的“一次编写，各处运行”的优点。

JDBC 的优点表现在如下几个方面：充分利用已有的数据资源，使用 JDBC, Java 程序可以很方便的访问已有的数据库资源，这使得我们可以引入 Java 技术，而不必丢弃

以前的系统和数据，这对 Java 技术的推广十分重要。其次，开发简便，JDBC API 比较简单，这使得使用 Java 开发数据库应用系统十分方便。其三，零配置，JDBC 的一个突出优点是 JDBC 本身无需任何配置，JDBC API 已经是 Java SDK 的标准组成部分，因此安装了 Java SDK，就安装了 JDBC。在 JDBC 中，所有建立数据库连接所需的信息都通过 JDBC URL 定义，也不需要任何配置。

2.5.2 JDBC 的体系结构

(1) JDBC 的组成与工作机制

JDBC 是实现 Java 应用程序与各种不同数据库对话的一种机制。JDBC 由两部分与数据库独立的 API 组成，一部分是面向程序开发人员的 JDBC API，另一部分是面向底层的 JDBC Driver API。JDBC 提供了一个通用的 JDBC Driver Manager，用来管理各种数据库软件商提供的 JDBC 驱动程序，从而访问其数据库。此外，对没有提供相应 JDBC 驱动程序的数据库系统，开发了特殊的驱动程序：JDBC-ODBC 桥，该驱动程序支持 JDBC 通过现有的 ODBC 驱动程序访问其数据库系统。JDBC 的基本层次结构由 Java 程序、JDBC 驱动程序管理器、驱动程序和数据库四部分组成。

Java 程序：Java 程序包括 Java 应用程序和 Java 小应用程序，主要是根据 JDBC 方法实现对数据库的访问和操作。其主要任务有：请求与数据库建立连接；向数据库发送 SQL 请求；为结果集定义存储应用和数据类型；查询结果；处理错误；控制传输、提交及关闭连接等操作。

JDBC 驱动程序管理器：它能够动态地管理和维护数据库查询所需要的所有驱动程序对象，实现 Java 程序与特定驱动程序的连接，从而体现 JDBC 的“与平台无关”这一特点。其主要任务有：为特定数据库选择驱动程序；处理 JDBC 初始化调用；为每个驱动程序提供 JDBC 功能的入口；为 JDBC 调用执行参数等。

驱动程序：驱动程序处理 JDBC 方法，向特定数据库发送 SQL 请求，并为 Java 程序获取结果。在必要的时候，驱动程序可以翻译或优化请求。使 SQL 请求符合 DBMS 支持的语言。其主要任务有：建立与数据库的连接；向数据库发送请求；用户程序请求时，执行翻译；将错误代码格式化成为标准的 JDBC 错误代码等。JDBC 是独立于数据库管理系统的，而每个数据库系统均有自己的协议与客户机通信，因此，JDBC 利用数据库驱动程序来使用这些数据库引擎。JDBC 驱动程序由数据库软件商和第三方软件商提供，因此，根据编程所使用的数据库系统不同，所需要的驱动程序也有所不同。

数据库：指 Java 程序需要访问的数据库及其数据库管理系统。

(2) JDBC 驱动程序的分类

JDBC 驱动程序实现在 JDBC API 中定义的所有抽象类和接口，为通用的 Driver Manager 提供 JDBC API。JDBC 驱动程序通常由数据库厂商提供，目前主流数据库产品都提供相关的 JDBC 驱动程序。根据访问数据库的技术不同，JDBC 驱动程序相应地分为四种类型。不同类型的 JDBC 驱动程序有着不一样的特性和使用方法。

① JDBC-ODBC 桥驱动程序 (JDBC-ODBC Bridge)。此类驱动程序提供了通过 ODBC 驱动程序的 JDBC 访问。其特点是必须在本地计算机上先安装好 ODBC 驱动程序，然后通过 JDBC-ODBC Bridge 的转换，将 Java 程序中使用的 JDBC API 访问指令转换成 ODBC API 指令，进而通过 ODBC 驱动程序调用本地数据库驱动代码完成对数据库的访问。这种方法的不足是：执行效率比较低，不适合对大数据量存取的应用；要求客户端必须安装 ODBC 驱动。

② 部分 Java 的本地 JDBC API 桥驱动程序 (JDBC-NativeAPI Bridge)。同第一类一样，此类驱动程序也必须在本地计算机上先安装好特定的驱动程序（类似 ODBC），然后通过 JDBC-Native API Bridge 的转换，把 Java 程序中使用的 JDBC API 转换成 Native API，进而存取数据库。这种方法效率比第一类驱动程序效率虽然高一些，但仍然需要在每台客户机上预先安装本地 API 库，因此不利于维护和使用。

③ 纯 Java 的 JDBC 中间件驱动程序 (JDBC-Middleware)。使用这类驱动程序时不需要在本地计算机上安装任何附加软件，但是必须在安装数据库管理系统的服务器端加装中间件，这个中间件负责所有存取数据库时必要的转换。此类驱动程序能将 JDBC 访问转换成与数据库无关的标准网络协议（通常是 HTTP 或 HTTPS）送出，然后由一个中间件服务器再将其转换成数据专用的访问指令，完成对数据库的操作。中间件服务器能支持对多种数据库的访问。由于是基于中间件服务器的，这类驱动程序的体积最小，效率较高，具有最大的灵活性，缺点是需要一个中间服务器的支持。此类驱动采用标准的网络协议，可以被防火墙支持，是 Internet 应用理想的解决方案。

④ 纯 Java 的 JDBC 驱动程序 (Pure JDBC Driver)。使用这类驱动程序时无需安装任何附加的软件（无论是本地计算机或是数据库服务器端），所有存取数据库的操作都直接由 JDBC 驱动程序来完成。此类驱动程序能将 JDBC 调用转换成 DBMS 专用的网络协议。数据库厂商是这一类驱动程序的主要提供者。它允许从客户机到数据库服务器的直接调用。这种驱动程序的效率最高，但由于采用 DBMS 专用的网络协议，可能不被防火墙支持。在 Internet 应用中会存在潜在安全隐患。

综上所述，最佳的 JDBC 驱动程序类型是第四类，它不会增加任何额外的开销，并且由纯 Java 语言开发而成，拥有最佳的兼容性。由于第一类和第二类的 JDBC 驱动程序都必须事先安装其他附加的软件，有损 Java 数据库程序的兼容性。第三类 JDBC 驱动程

序也是不错的选择，它也是由纯 Java 语言开发而成的，并且中间件也仅需要在服务器上安装。因此，建议最好以第三类和第四类 JDBC 驱动程序为主要选择，第一类和第二类的 JDBC 驱动程序为次要选择。

2.5.3 JDBC 的关键接口和数据库访问

(1) JDBC API 的关键接口

JDBC 主要有两种接口，分别是面向程序开发人员的 JDBC API 和面向底层的 JDBC Drive API。在 JDBC API 定义了一些数据库操作的类和接口，通过这些接口实现与数据库连接，执行 SQL 语句并得到返回结果等功能。下面是其中关键的类和接口：

① `java.sql.Driver` 和 `java.sql.DriverManager`：管理驱动程序，为驱动程序的调入和创建新的数据库连接提供支持。

② `java.sql.Connection`：完成对某一指定数据库的连接功能。

③ `java.sql.Statement`：代表一个特定的容器，用来构造并执行 SQL 语句，实现对指定数据库的访问操作。它又分为两个子类：`java.sql.PreparedStatement` 和 `java.sql.CallableStatement`，前者用于执行预编译的 SQL 语句的声明，后者用于执行对一个数据库存储过程的调用。

④ `java.sql.ResultSet`：处理结果集，用来控制对一个特定 SQL 语句的行数据的存取。然而，JDBC Drive API 则必须提供对上述接口的支持，它是面向驱动程序开发商的编程接口。对于大多数数据库驱动程序来说，只要实现 JDBC API 提供的抽象的类就可以了。

(2) JDBC 数据库访问的一般算法

① 先安装好待访问的数据库的 JDBC 驱动程序，并在 ClassPath 环境变量中设置对它的检索路径。

② 在数据库访问程序中引入标准的 `java.sql` 包：`import java.sql.*`；

③ 调用 `Class.forName()` 方法加载相应数据库的 JDBC 驱动程序；

④ 调用 `DriverManager` 类的 `getConnection()` 方法创建 JDBC Connection 对象，以连接数据库；

⑤ 调用 JDBC 的 `Statement` 类的 `CreateStatement()` 方法，创建一个 `Statement` 类的对象（SQL 语句对象）；

⑥ 调用 SQL 语句对象的 `ExecuteQuery()` 方法，执行相应 SQL 语句，并获得返回的 `ResultSet` 对象；

⑦ 调用 `ResultSet` 对象的移动记录针等方法，对结果集进行相关操作；

⑧ 访问结束，则调用 `close()` 方法，释放 Java 程序与数据库的连接并关闭数据库。

在上述算法中可将 SQL 语句完整直接的写在 Java 程序中，实现对数据库的访问，这种访问模式属于静态 SQL 模式，它的灵活性较差，且受到调用级的接口的限制而只能执行基本的 SQL 语句。然而，也可以在 Java 程序运行时提交一个由数据库服务器处理的 SQL 请求，这种机制也意味着动态编译、授权和权限检查，从而使程序可以灵活地在运行时定义可变的查询，这种访问模式被称为动态 SQL 模式。

3 系统需求分析

3.1 系统功能需求

鞍山师范学院计算中心主要承担学院非计算机相关专业的计算机公共课教学任务，例如《计算机应用基础》、《高级语言程序设计》等公共课程。所以，如果采用传统的考试方式，在安排期末考试的时候，由于庞大的考生数量，不但考场安排相对困难，还要花费大量人力物力去印刷、分发和保管考生卷纸以及成绩。而且，采用笔试的考察方式，使得教师在出题的时候，受到了相当大的限制，考试结果也不能够体现考生的实际操作能力。为了适应学院依托校园网进行现代化管理和教学的需要，切实、有效地做好期末考试的各项管理工作，解决以上种种问题，计算中心采用网络考试系统，实现考试的无纸化管理。

系统应采用开放、动态的系统架构，将传统的考试模式与当前在高校中普遍应用的校园网相结合。采用计算机在线考试的方式，并且，在考试之前，首先要由授课的教师建立试题库，并设置好相应的试题参数。然后，在考生进入系统前应该对考生进行身份验证，以避免成绩记录出错。考生进入考试系统后，也应能根据安排，选择相应的考试科目。同时，由于考场一般安排在计算中心所管理的各个机房，考生之间的距离相对较近，为了保证考试过程的规范和公平，对于相邻考生来说，试卷的题型、题量和难度都应是相同的，但试题并不相同。

试卷是考试中的关键要素，而在线考试基于网络环境，试卷应该从服务器的试题库随机抽取试题后动态生成的。考虑到系统的实用性，从试卷内容设计上，题型要求多样化，既要有选择题、判断题等客观题，又要有填空题、简答题等主观题。所以一定要考虑各种不同的题型在试题库中的保存方式及主观题和客观题的评分问题。另外，系统还应该对考试时间进行严格控制，考试结束时要求考生交卷。考试者选择提交后，考试系统应该能够提供自动阅卷功能，并将成绩保存。考试系统还应当根据考试情况对试题区分度、难度进行评估，并对考试成绩进行综合分析比较。此外，相关的考试管理人员还应该能够方便、快捷地对在线考试系统及各项数据的进行日常的维护和备份。

依据对在线考试系统的以上要求，并依照相关考试的具体流程，考试系统的交互界面应简洁、友好，并具有很强的安全性、稳定性及以下基本功能：

(1) 用户身份验证：考生在登录系统之后，应该对考生的输入的身份信息进行核对，并将结果反馈给考生。

(2) 试题库管理：试题库中存储着所有的考题及其相关的试题属性，包括题目所属的知识点、章节、题目的难度等。各授课教师在登录试题库后，能够对题库进行添加题目、编辑题目、删除题目等维护更新工作。

(3) 组织考试：根据统一安排，在规定的考试时间开放考试系统。使考生能够及时登录到考试系统，经过系统基考生本人的确认后，开始答卷。系统在醒目位置提示考试所剩时间，考试结束，系统能够自动终止考生答卷并收取试卷以备批改。

(4) 自动组卷：组卷是考试的关键，系统应根据考试所需的题型和题量，按照相应的组卷规则，从试题库里抽取试题生成合乎要求的考卷。

(5) 自动阅卷：由于客观题的答案相对固定，所以系统应能够自动对考生的客观题答案进行评分，而主观题部分则交由教师人工阅卷，考生的总成绩由计算机自动计算，并将结果保存在数据库中。

(6) 综合检索：检索功能应根据用户的身份，分配相应的检索权限和检索范围，对试题、试卷、成绩和个人信息等进行查询，并根据权限进行相应的更新。

(7) 考试分析：考试系统在考试结束后，应根据考生的答卷情况及成绩的进行综合的比对和分析，分析结果也能够用于对题库中题目的区分度、难度的重新估算。

3.2 系统用例分析

3.2.1 识别系统参与者和用例

在分析了系统功能后，采用面向对象的设计思想，借助统一建模语言（UML）中的用例图（Use Case Diagram）来描述考试系统的总体功能需求。

用例图包括两个主要元素：用例（Use Case）和参与者（Actor）。其中，用例体现了系统中的各功能模块，用例是系统为响应由参与者所发起的一个事件序列而执行的相应处理。参与者也可称作“系统用户”，既可以是使用系统的相关人员，也可以是另一个系统或设备，甚至可以是某个时间段，参与者能帮助开发者界定系统的功能边界。所以，在需求分析阶段，用例图是处于应用领域的层次上，是最高层次的模型，即用例图不需要涉及系统实现的相关细节，此时，用例图只关注系统要完成何种功能，而不关注系统具体要怎样才能完成该功能。

具体到本在线考试系统，其主要的目标是实现在线考试，围绕考试这一中心目标，系统应提供对考试过程的全面支持并实现其它辅助功能，例如自动组卷，自动评卷、人工评卷、成绩查询、试题库管理、成绩管理等等。

在获取用例之前，首先要确定系统的参与者。通过前面的系统功能分析可知考试系统的用户主要有三类：考生、教师和管理员，他们具有不同的权限等级，在进入考试系

统前，都必须经过系统的身份验证，而且，只有是合法的用户，才能够进入考试系统，并按照相应的权限对规定系统资源进行访问。所以，考试系统的参与者可以确定为：考生（Student）、教师（Teacher）和管理员（Administrator）。同时，因为参与者也是类，所以多个参与者之间可以具有与类之间相同的关系。在用例图中，往往有的参与者既扮演着自身的角色，却也同时扮演了另外的更具一般化的角色，那么就可以用泛化关系来描述它们。在考试系统中，教师和管理员就是这样的关系。在确定了系统的参与者之后，就可以根据这三类参与者，依次识别考试系统的各相关用例。

(1) 与考生（Student）相关的用例

① 在线考试：作为考试系统的核心功能，首先要为考生提供试卷，并对考生的答卷过程进行倒计时，到达规定时间，能够自动收取试卷，并对其中的客观题进行自动评分，然后记录成绩。

② 查询成绩：考生能够根据自己的姓名、考号等个人信息查询考试成绩。

③ 浏览公告：能够浏览由管理员发布在网上的各项公告，例如考试指南、考试时间、考场安排、监考安排、成绩发布等。

(2) 与教师（Teacher）相关的用例

① 更新题库：教师可以按科目查询、添加、修改、删除试题库中的题目。

② 管理试卷：教师根据讲授的教学内容和教学进度，设定相应的自动组卷规则。另外，还要求在必要的情况下，能够从题库中人工选择试题并组成试卷。

③ 批改试卷：主要指批改试卷中的主观题，也能够对自动批阅的客观题进行核对。

④ 成绩汇总：教师可以查询某班级某门科目，考生的所有成绩，并可以对成绩进行综合分析，例如班级、院系的最高分、最低分、平均分，以及各个分段的人数等。

⑤ 浏览公告：能够浏览由管理员发布在网上的各项公告，例如考试指南、考试时间、考场安排、监考安排、成绩发布等。

(3) 与管理员（Administrator）相关的用例

① 用户管理：管理员可以对系统中的所有用户的个人信息进行诸如增加、删除、修改等操作。例如，如果考生已经毕业，那么应将其在系统中的相关记录删除或备份，同时，为刚刚入校的新生建立信息记录，并更新院系、班级以等信息。

② 考试管理：在考生参加考试之前，管理员需要统一安排各院系的考试时间、考试地点和监考教师，并发布公告。而在考生考试的过程中，管理员还应该能够随时掌握各考场的考试人数和交卷情况。

③ 更新公告：更新于考试相关的各项公告，如考场、监考教师安排。

④ 系统维护：对考试系统进行日常管理并及时处理常见问题，并对系统中的用户信息、试卷、成绩、日志等重要数据进行管理和备份。

3.2.2 用例建模

用例模型由用例、参与者和关联线共同组成。用例可大可小，既可能大到覆盖了整个系统，也可能是子系统或者系统中的一个组件，但是，无论大小，用例必须能够完整地描述用户所需要实现的一个具体功能。所以，用例建模一般先要确定好系统的参与者，然后以每一个参与者所期望的行为或需要系统提供的行为作为基础进行建模。建模时，首先要从顶层用例开始，然后对顶层用例逐步分解细化，直至得到用例图能够清晰的描述系统各功能需求为止。

根据前面的分析，可见容易地建立起顶层的用例模型，如图 3.1 所示。在系统层，用例表示整个系统对参与者可见的行为，每个用例的执行都独立于其他用例。在用例模型中，每个参与者都可以参与一个或多个用例。它通过交换信息与用例发生交互（因此也与用例所在的子系统发生了交互），而参与者的内部实现是与用例不相关的。

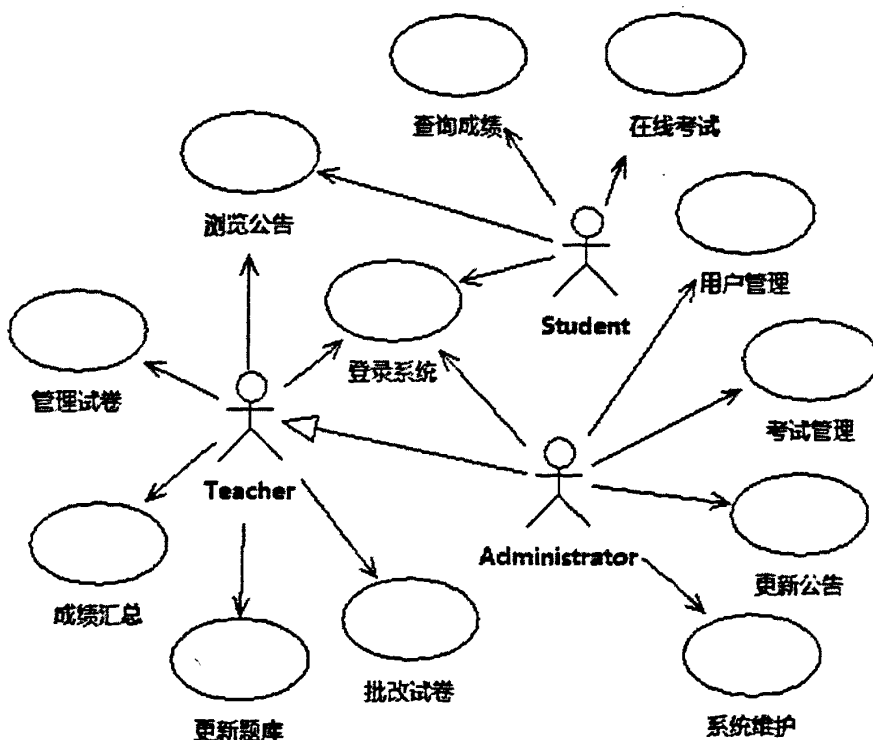


图 3.1 考试系统用例关系图

Fig. 3.1 Use case relation diagram of system

上图中所绘制的顶层用例模型，是从整个系统的层次上对考试系统的各主要功能进行描述。而对于一个复杂的考试系统，还需要通过分层分解，逐步细化的方法来描述系统的功能。限于篇幅，在本节中只重点介绍考试系统中的两个主要用例的细化分解，即考生在线考试和教师管理题库这两个用例进行的细化分析。

当考试开始后，考生首先选择当次考试的考试科目，并得到系统从试题库中自动抽取的试卷，然后开始在线答卷，期间可以随时交卷，而当考试的规定时间结束后，系统会自主交卷，交卷后，系统首先对客观题进行自动阅卷，之后将试卷和成绩保存以备核对。所以，对考生在线考试用例分解后，可以得到选择考试科目、答卷、交卷、自动阅卷和保存成绩等几个细化用例，各个用例之间的关系如图 3.2 所示。

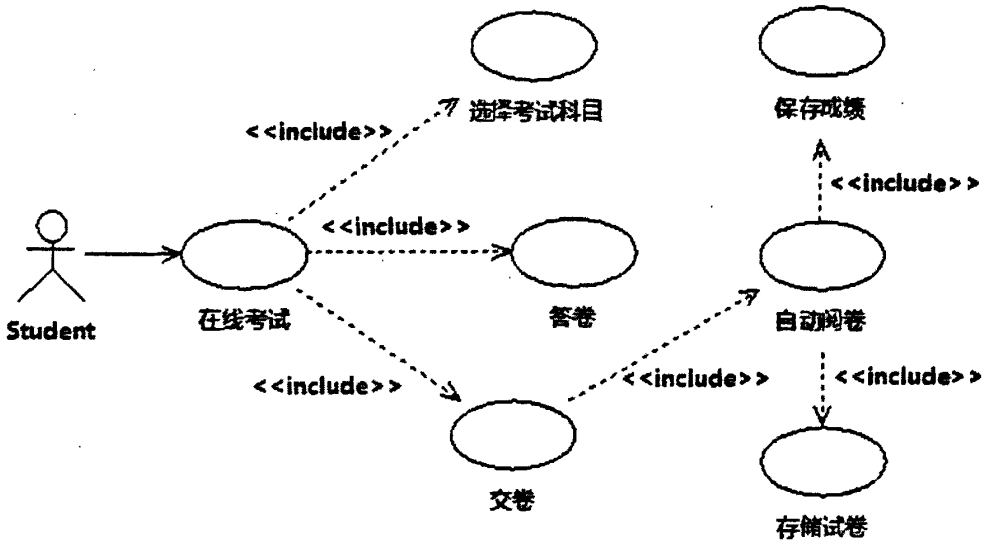


图 3.2 在线考试用例图

Fig. 3.2 The use case diagram of online exam

对于教师管理题库用例，当教师通过系统的身份验证，成功登录考试系统后，系统根据教师所教授的课程，分配相应的权限使登录教师可以查看题库中的相关试题，并能够对试题进行增加、删除和修改等管理操作，也可以仅仅浏览题库中的题目。而在进行添加试题或者修改试题之后，系统能够将刚才更新过的题目再次显示给教师以备核对。所以，更新题库用例可以进一步细化为添加试题、查看试题、修改试题和删除试题等几个用例。细化后的教师更新题库用例图如图 3.3 所示。

测到这个文件，并将其解压。在浏览器中访问这个应用的 JSP 时，通常第一次会很慢，因为 Tomcat 要将 JSP 转化为 Servlet 文件，然后编译。编译以后，访问将会很快。

(3) 数据库：SQL Server 2000

由于 SQL Server 具有强大的数据管理与处理能力，具有较高的安全性能^[24]；每个数据库中可创建多达 20 万个数据表，数据表中的记录的行数只受服务器硬盘空间的限制^[25]；支持远程存取等，符合考试系统的需求，因此选择 SQL Server 2000 作为后台数据库。

(4) JSP 开发工具：JBuilder

系统选用工 Borland 公司的 JBuilder 作为开发工具。JBuilder 软件的目标定位是代码开发人员而不是高级设计人员，所以 JBuilder 中包含了大量的向导程序和其他针对中间层的快速开发工具。JBuilder 性能稳定、使用方便，特别适用于创建 Java 组件。此外该软件还提供了很多 JSP 功能。

3.3.2 运行环境

(1) 服务器端

一般情况下，服务器端的最低配置是由建立站点所需的软件来决定的，但在使用最低配置的情况下，服务器的稳定性往往不尽如人意。同时，现在的硬件性能已经相当出色，而且价格也相对便宜，因此，在条件允许的情况下，可以尽量应给服务器端配置更高性能的硬件。

本系统服务器端的采用的操作系统为 Windows 2000 Server，Web 服务器采用 Tomcat 5.0，数据库为 SQL Server 2000。

(2) 客户端

对于客户端，在线考试用的计算机只须安装操作系统和浏览器，在接入网络后，无须作任何其他配置即可使用本在线考试系统。

4 系统设计

4.1 考试系统的类图

在UML中,类图(Class Diagram)是描述类、接口、协作以及它们之间关系的图,用来显示系统中各个类的静态结构。通过分析用例,就可以得到相关的类,类是面向对象系统组织结构的核心。确切的说,类图即是根据系统中的类以及各个类之间的关系去描述系统的静态视图。类图是面向对象系统建模中最常用的图,也是进行其他建模的基础。类图主要包括类、对象和它们之间的关联等主要元素。可将所有的类分为实体类、边界类和控制类三种。

实体类是系统中的核心类,用于长期保存系统中的信息,以及针对这些信息的相关处理行为,实体类的对象和应用系统本身具有相同的生命周期。由于实体类一般是与显示事务相对应的类,所以可以映射成用于数据存储的各种数据库或数据表,在考试系统中考生类、教师类、试题类等都是实体类。

边界类是从与系统和外界进行交互的对象中归纳和抽象出来的,边界类是系统内的对象和系统外的参与者之间的联系媒介,外界的消息只有通过边界类的对象才能发送给系统。所以,一般来说,系统中的边界类即指用户与系统间的交互界面,例如登录页面、查询页面、考试界面、题库管理界面等。

控制类用于管理实体对象与边界对象之间的消息传递,即协调系统内边界类与实体类之间的交互。每个用例通常都有一个控制类,用来控制用例中的事件顺序,控制类也可以在多个用例之间公用。在考试系统中,控制类主要有在线考试、成绩查询、考生管理和试卷批阅等控制类。下面主要对与在线考试关系较为密切的实体类进行介绍。

与用户信息相关的类有考生类(Student),教师类(Teacher)和管理员类(Administrator)。这三个类的都有密码(password)、姓名(name)等基本属性。此外,考生类的属性有考号(studentID)、专业编号(majorID)、年级(grade)、班级(classID)等。教师类的属性有教师编号(teacherID)、教授科目(courseID)等。

考试系统中还有专业类(Major)和科目类(Course),专业类的主要属性有专业名称(majorName)、班级编号(classID);科目类主要属性有科目名称(courseName)和科目编号(courseID)。

在试题与试卷方面的实体类主要有试题库类(Question)、科目类(Course),试卷信息类(PaperInfo)、试卷内容类(PaperDetail)、答案类(ExamAnswer)和考试成绩类(ExamScore)等。

其中，试题类的主要属性有试题编号 (itemID)、题型种类 (itemType)、科目编号 (courseID)、题目内容 (content)、标准答案 (answer)、题目分值 (value) 和难度 (level)。试卷信息类的主要属性有考生考号 (studentID)、试卷编号 (paperID)、和科目编号 (courseID)。试卷内容类的主要属性有试卷编号 (paperID)、试题编号 (itemID)。试卷答案类的属性主要有试卷编号 (PaperID)、试题编号 (itemID)、考生考号 (studentID) 和考生答案 (content)。成绩类的属性有考号 (studentID)、总分 (totalScore) 和科目编号 (courseID) 等。

图 4.1 为与考生和考试有密切关系的实体类图 (类中的属性和方法也做了简化)。

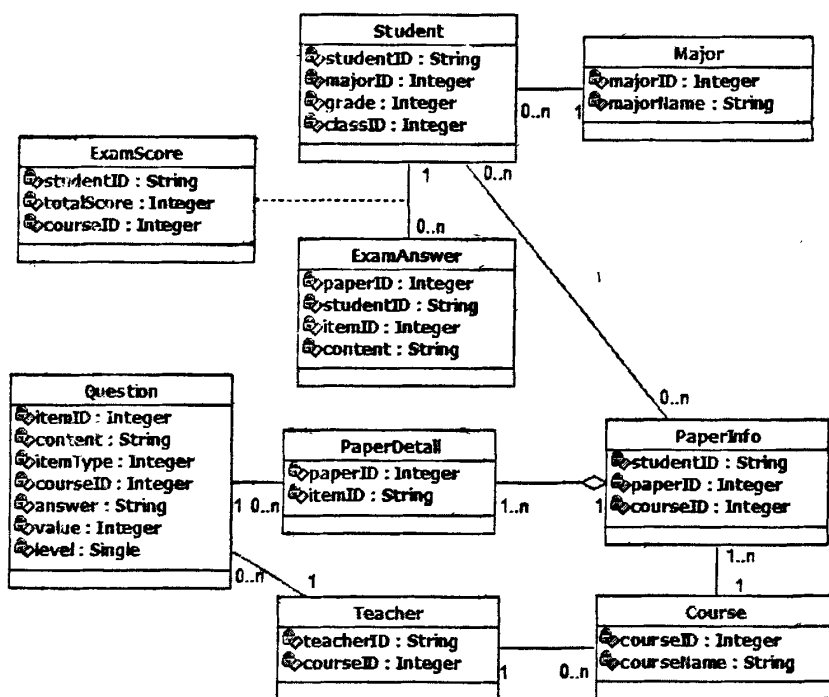


图 4.1 系统类图

Fig. 4.1 The class diagram of the system

4.2 系统中主要功能模块的详细设计

在系统的详细设计阶段，可以使用顺序图来描述之前在需求分析过程中所得到的各个用例的实现过程。顺序图对与系统设计十分重要，顺序图是以时间为序，描述了对象以及对象之间的消息传递，当系统执行一个用例行为时，时序图中的每条消息对应了一

个类操作或状态机中引起转换的触发事件。顺序图是交互图的一种，主要包含了四种元素，分别是对象、生命线、消息和激活。顺序图表示为二维图，纵轴是时间轴，时间沿竖线向下延伸。横轴代表了各个独立的对象。当对象存在时，生命线用一条虚线表示，当对象的过程处于激活状态是，生命是一个双道线。消息用从一个对象的生命线到另一个对象生命线的带箭头的消息线表示，并且消息线是以时间为序在图中从上到下排列在顺序图中。限于篇幅，这里主要列出了用户管理、试题管理和在线考试等几个用例的实现设计。

4.2.1 用户管理

用户管理主要是对考生与考试相关的个人信息进行管理。考生一般按照专业进行划分。所以，更新考生信息时，首先从考生管理页面中选择考生的专业，考生管理页面会向考生管理控制类提交所选的专业，用户管理控制类再据其从专业类中所获取的专业编号，从考生类中检索出属于该专业的所有考生的个人信息，并在考生管理页面上显示所有的检索结果以供管理员查看。这时系统管理员可以根据需要在考生管理页面中进行增加、修改或者删除所选考生信息等操作，而用户管理控制类会根据用户管理页面向其发送的消息进行相应的更新操作，处理结束后会将更新后的用户信息重新保存到考生类当中，并返回更新结果到考生管理页面上供管理员查看核对。顺序图如图 4.2 所示。

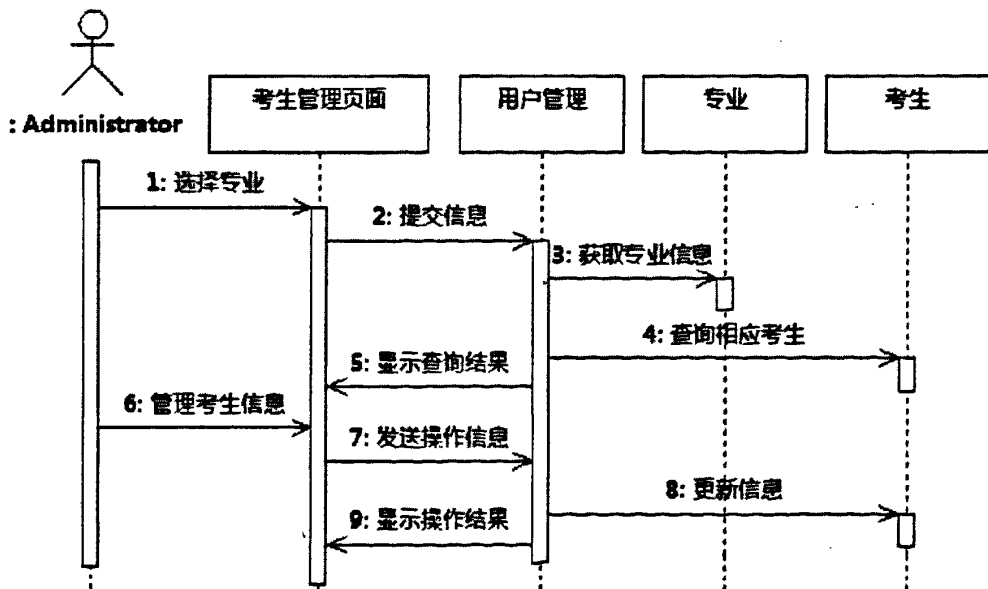


图 4.2 考生管理顺序图

Fig. 4.2 The sequence diagrams of student management

4.2.2 更新题库

教师在进行题目管理之前，要先从题库中查询出需要进行更新的题目。所以，教师首先要在题库管理页面中输入有关题目的查询条件，例如题目的题型、题号、难度等。题目查询控制类在得到查询条件后从是题库中获取所有符合条件的题目，并显示在题目管理页面上。

随后，对试题库进行管理操作，例如添加题目，修改题目或删除题目，这些操作由题目管理控制类发送消息给题库，完成相应的更新操作后，系统会将更新后的题目显示给教师。题目管理的顺序图如图 4.3 所示。

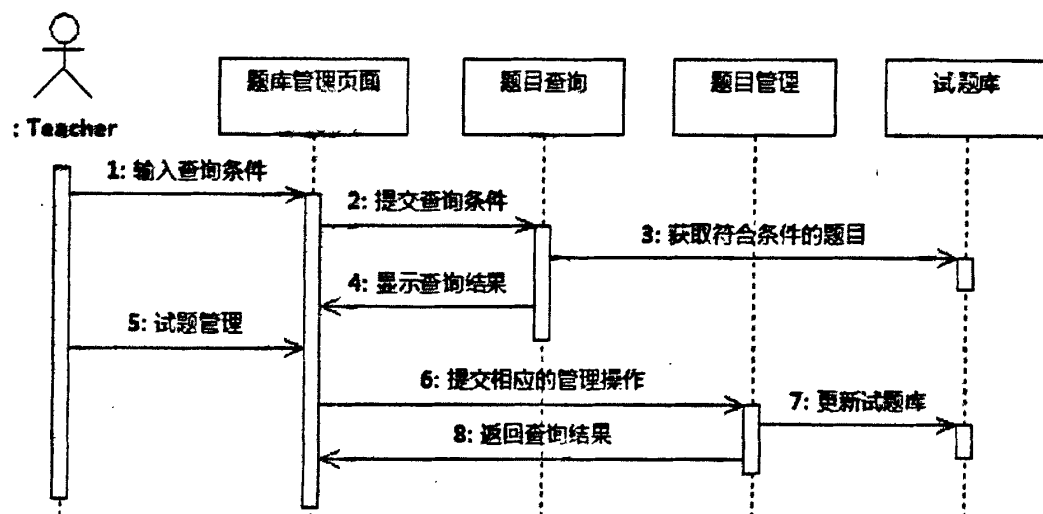


图 4.3 更新题库顺序图

Fig. 4.3 The sequence diagrams of update question database

4.2.3 在线考试

考生在线考试是系统中的主要功能。当考生登录考试系统后，首先要从考试页面选择考试科目，考试控制类在收到考试页面发送的科目信息后会从科目类中检索出相应的科目编号，然后发送科目编号给组卷控制类，组卷控制类会根据相应的出题策略及考试科目从试题库中抽取题组成试卷，并返回给考生。考生在得到试卷后开始答卷，结束答卷后向考试系统提交试卷，这时，考试控制类会首先保存考生的答卷，然后进行自动阅卷并保存成绩。最后，向考生返回考试结束的提示信息。

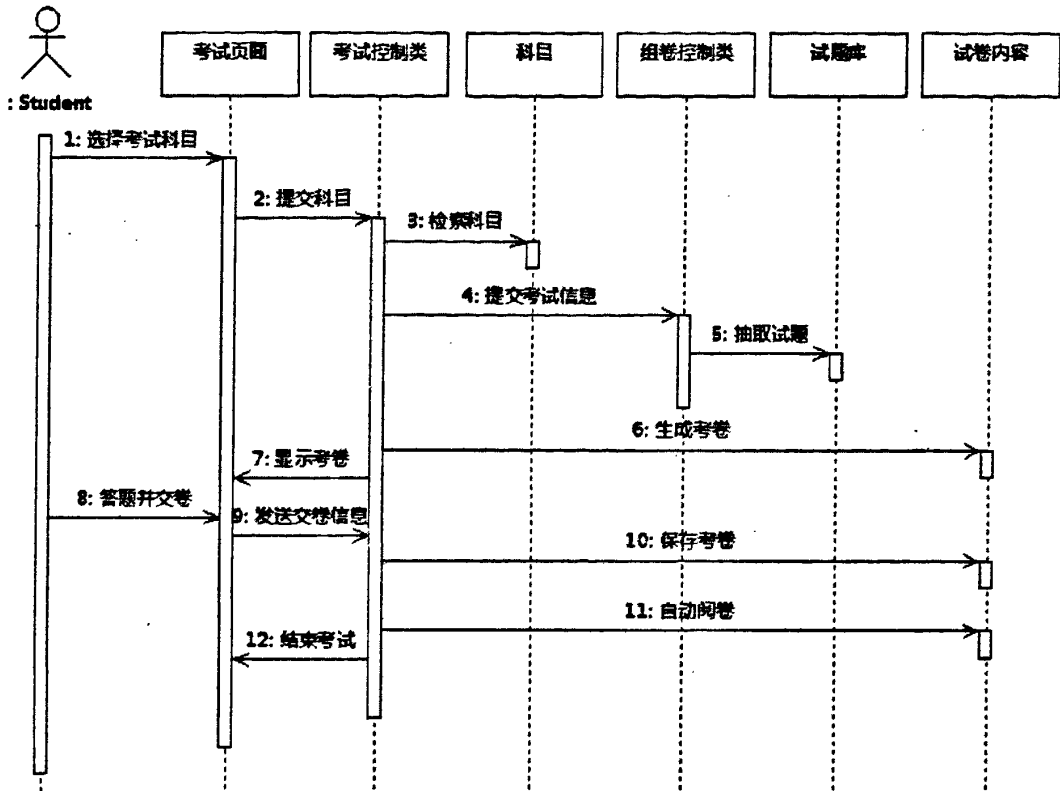


图 4.4 在线考试顺序图

Fig. 4.4 The sequence diagrams of exam

4.3 数据库设计

数据库技术是现代信息科学与技术的重要组成部分，是计算机数据处理与信息管理系统核心。而随着计算机技术与网络通信技术的发展，数据库技术已成为对大量数据进行组织与管理的重要手段及软件技术，是网络信息化管理系统的基础。所以，网上考试系统离不开数据库系统的支持，一切与考试相关的数据，如题目、答案、成绩和考生信息等等都要分门别类的存放到系统的数据库当中。同时，数据库中的数据独立于应用程序，包括数据的物理独立性和逻辑独立性，都给数据库的使用、调整、优化和扩充提供了方便，进一步提高了数据库应用系统的安全性和稳定性。

所以，数据库的设计是进行系统设计的关键，数据库也是整个系统运行的核心。通过前面的需求分析，了解到了系统的各项用户需求后，可知考试系统中主要有如下的数据项和数据结构：

- (1) 考生信息：系统用户名、密码、姓名、考号、专业、年级、班级等。

- (2) 专业：专业名称、编号。
- (3) 试题库：题目内容、答案、分数、题号、科目、章节、知识点等。
- (4) 考试科目：科目名称、科目编号。
- (5) 试卷内容：试卷编号、题目编号。
- (6) 考生答案：考生考号、试卷编号、题目编号、考生答案和得分等。
- (7) 成绩：试卷编号、考生考号、考生成绩。

4.3.1 数据库的概念结构设计

数据库的概念结构设计是将需求分析中得到的用户需求抽象为信息结构及概念模型的过程，数据库的概念结构能够真实的反映现实世界中的事务及事务之间的联系。在数据库的概念设计中，一般采用 E-R 数据模型来表示概念设计表示模型。通过上面对考试系统各数据项和数据结构的分析，以及对现实世界概念模型的抽象，可以确定在线考试系统中，主要有教师、考生、题库、试卷等实体。数据库的 E-R 图如图 4.5 所示。

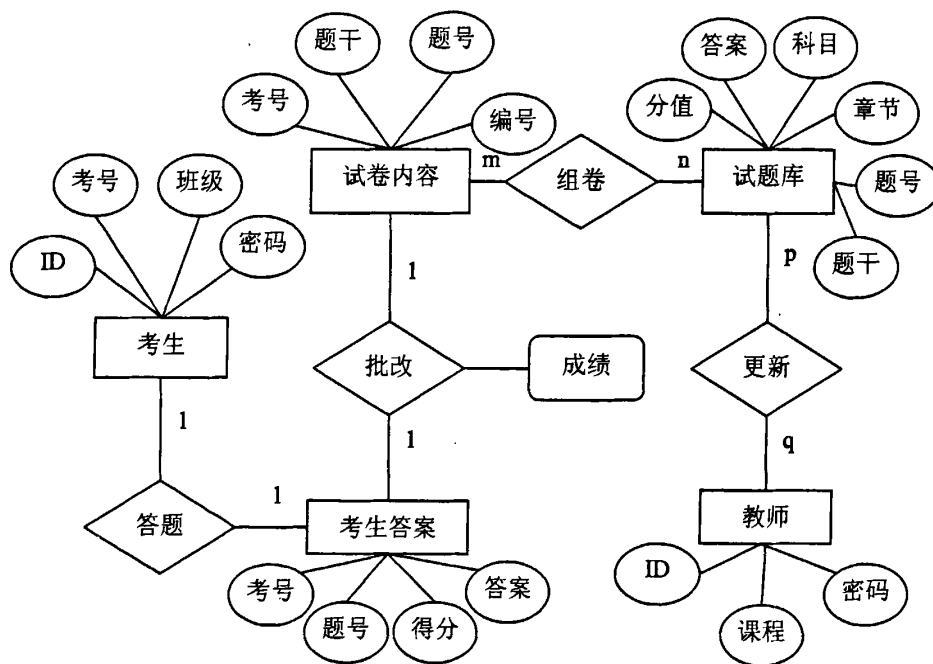


图 4.5 系统中主要实体 E-R 图

Fig. 4.5 Primary E-R rational diagram

4.3.2 数据库的逻辑结构设计

根据实际需要,在线考试系统选用了微软公司的 SQL Server 2000 数据库系统。而数据库的逻辑设计是将数据的概念结构转化为 SQL Server 2000 数据库系统实际模型^[26],从而得到数据库的逻辑结构,以便在数据库中建立表结构。考试系统中的各主要数据表如以下表格所示。

表 4.1 题库信息表

Tab. 4.1 Question database information table

字段名称	说明	数据类型	长度	允许空
itemid	题目编号	int	4	N
item_type	题型	int	4	N
courseid	课程编号	int	4	N
chapter	章节	int	4	N
content	题目内容	text	16	N
teacherid	出题教师	varchar	10	N
ref_key	答案	text	16	N
difficulty	难度	int	4	N
knowledge	知识点	int	20	N
knowlevel	掌握程度	varchar	10	N
dist	区分度代码	int	4	N
score	题目分值	int	4	N
frequency	试题被抽取次数	int	4	N

表 4.2 试卷内容表

Tab. 4.2 Paper detail table

字段名称	说明	数据类型	长度	允许空
paperid	试卷编号	int	4	N
itemid	试题编号	int	4	N

表 4.3 考生成绩表

Tab. 4.3 Score of exam table

字段名称	说明	数据类型	长度	允许空
stuid	考生考号	varchar	20	N
examid	考试编号	int	4	N
score	考试成绩	int	4	N

表 4.4 考试状态表

Tab. 4.4 State of exam table

字段名称	说明	数据类型	长度	允许空
state	考试状态代码	int	4	N
remark	代码所对应的状态	varchar	10	N

表 4.5 考生答案表

Tab. 4.5 Answer table

字段名称	说明	数据类型	长度	允许空
stuid	考生学号	varchar	20	N
examid	考试编号	int	4	N
itemid	题目编号	int	4	N
answer	考生答案	text	16	N
score	考生得分	int	4	N

表 4.6 考生信息表

Tab. 4.6 Student information table

字段名称	说明	数据类型	长度	允许空
loginname	用户名	nvarchar	10	N
password	登陆密码	nvarchar	10	N
user_type_id	用户类型编号	int	4	N
is_forbid	是否禁止登录	bit	1	N
stuid	考生学号	varchar	20	N
major	专业	varchar	10	N
grade	年级	int	4	N
class	班级	int	4	N

表 4.7 教师信息表

Tab. 4.7 Teacher information table

字段名称	说明	数据类型	长度	允许空
loginname	用户名	nvarchar	10	N
password	登陆密码	nvarchar	10	N
user_type_id	用户类型编号	int	4	N
is_forbid	是否禁止登录	bit	1	N
teacherid	教师编号	varchar	10	N
courseid	教授科目	int	4	N

4.3.3 数据库关系图

数据库中的关系图如下图所示：

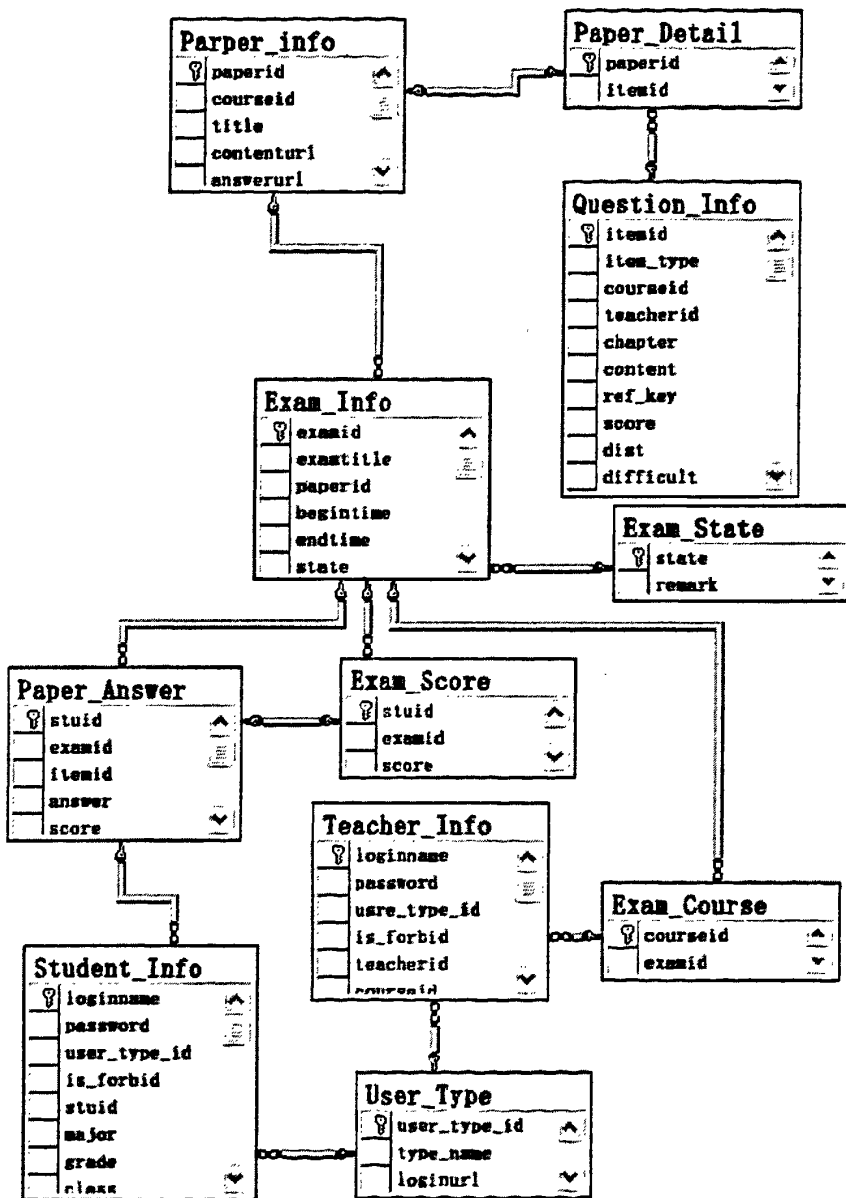


图 4.6 数据库的关系图

Fig. 4.6 The relation of the database tables

5 考试系统中的组卷算法

5.1 遗传算法概述

5.1.1 遗传算法的基本概念

遗传算法 (Genetic Algorithm, GA) 是模拟达尔文的遗传选择和自然淘汰的生物进化过程的计算模型, 是一种通过模拟自然进化过程搜索最优解的方法^[27]。所以, 遗传算法吸取了自然界中“适者生存, 优胜劣汰”的进化理论, 为解决许多传统的优化方法难以解决的优化问题提供了新的途径。由于遗传算法的整体搜索策略和优化搜索方法在计算中不依赖于梯度信息或其他辅助知识, 而只需要影响搜索方向的目标函数和相应的适应度函数, 所以遗传算法提供了一种求解复杂系统问题的通用框架, 它不依赖于问题的具体领域, 对问题的种类有很强的鲁棒性。如今, 遗传算法不论是在算法设计上还是在基础理论上, 均已取得了长足的发展, 已成为信息科学、计算机科学、运筹学和应用数学等诸多学科所共同关注的热点研究领域^[28]。

遗传算法作为一种概率搜索算法, 借鉴了生物学中自然选择和遗传机制的高度并行、随机、自适应的性质, 它利用某种编码技术作用于被称作染色体的二进制数据串, 其基本思想是模拟由这些染色体组成的群体进化过程。由于遗传算法是由进化论和遗传学理论相结合而产生的直接搜索优化算法^[29], 因此, 在遗传算法中也借鉴了许多生物学中的术语。

(1) 个体 (Individual): 也称基因型个体, 个体是遗传过程中带有遗传特征的实体, 也是遗传算法中的所处理基本对象和结构。

(2) 基因 (Gene): 基因是携带遗传信息的基本单位, 用于表示个体的特征。

(3) 位串 (String): 与遗传学中的染色体的概念相对应, 是个体的表现形式。

(3) 种群 (Population): 一定数量的个体的集合叫做种群。

(5) 群体规模 (Population Size): 在群体中个体的数量称为群体大小。

(6) 适应度 (Fitness): 适应度表示某一个体对于生存环境的适应程度, 对于生存环境适应程度较高的个体将获得更多的繁殖机会, 而对生存环境适应程度较低的物种, 其繁殖的机会就会相对减少, 甚至逐渐灭绝。

(7) 遗传操作 (Genetic Operation): 遗传算法是一种群体型操作, 该操作以群体中的所有个体为对象。遗传操作由选择 (Selection)、交叉 (Crossover) 和变异 (Mutation) 三个基本操作算子组成。

① 选择：根据遗传学的理论，对生存环境适应程度较高的物种遗传到下一代的机会相对较高。所以在遗传算法中，应用选择算子对群体中的个体进行优胜劣汰操作，父代中适应度较高的个体被遗传到子代群体中的概率较大，而适应度较低的被遗传到子代群体中的概率较小。

② 交叉：遗传算法中的交叉算子使得在原始群体中的优良个体的特性能够在一定程度上继续得到保持，而另一方面，又使得算法能够探索新的基因空间，使新的群体中的个体更加多样性。

③ 变异：变异算子能够对群体中个体串的某些基因位置上的基因值作变动。遗传算法中，变异算子的应用使得算法具有了局部的随机搜索能力，并且可以使遗传算法始终维持群体的多样性。

5.1.2 遗传算法的流程

遗传算法的运行过程是一种的典型的迭代过程，遗传算法在整个进化过程中的遗传操作是随机的，但是算法能够有效地利用历史信息来预测下一代期望性能有所提高的寻优点集。所以，在这一不断进化的过程中，群体中的个体地得以逐代优化，并逐渐地收敛到一个最适应环境的个体上面，即获得最优解。以下为遗传算法的一般流程。

(1) 选择编码策略，将参数集合转换成位串结构空间。

(2) 定义适应度函数和遗传策略，并计算交叉概率、变异概率等遗传参数。

(3) 采用随机初始化的方式生成初始群体。

(4) 计算群体中的个体在通过位串编码后的适应度函数值。

(5) 依据设定好的遗传控制参数，使选择、交叉和变异算子作用于群体，产生下一代的群体。迭代执行，直到群体性能满足需要。

根据算法的流程，在遗传算法的运行过程中，对自变量的编码、初始群体的设定、适应度函数的设计、遗传操作（选择、交叉和变异）、遗传控制参数（包括群体规模、执行遗传操作的概率等）的设定是遗传算法中的核心内容。

5.2 基于遗传算法的组卷算法

一份高质量的试卷，应该在题型、难度、区分度和知识点分布等各项指标之间达到相对平衡。并能够最大程度满足用户要求。所以，组卷问题实际上是一个复杂的多目标组合优化问题，问题的求解精度越高，表明试卷质量越好^[30]。而传统的组卷方法往往很难解决这个问题，甚至很难描述这样复杂的带约束优化的问题。因此，选用一种合理组卷方法，可以保证系统自动生成的试卷能最大程度地满足出题者对试卷的需要，并使试

卷具有较高的随机性、科学性和合理性。此外，在对响应速度要求较高的网络交互环境下，算法的效率也是自动组卷的关键。

在考试系统进行自动组卷的时候，首先要将难度、知识点等相对模糊的要求进行量化，转化成计算机可以理解的要求，然后依据组卷算法从试题库中抽取一定数量并且满足要求的题目组成试卷。

5.2.1 组卷算法的数学模型

组卷问题的数学模型是组卷算法的基础。因此需要在分析组卷问题的基础上建立一个性能优良的数学模型。对于自动组卷功能来说，其要求实现的是通过获取用户的对试卷需求信息后，建立相应的试卷模式，然后根据该试卷模式建立组卷算法的数学模型。用户的需求对组卷系统来说是一种模糊的约束，因此，首先需要把这些模糊的约束量化成具体的并且能够被计算机识别的量化指标。

考试系统中的所有试题都被存放在试题库中，而题目本身也有其固定的属性，试题各项属性的确定能够直接影响到组卷系统的准确性和效率。试题的属性指标定量地描述了每一道试题的内在属性、外部特征以及它在考试测试中的功能，是计算机进行自动抽题组卷的基础。在组卷前，首先要给定的试卷的相关约束条件，例如卷面分数、难度系数、区分度以及不同题型、能力层次和知识点的题目所占比例等，并据其从大量的试题库中抽取出最优的试题组合。

所以，用 a_1 代表题目的分值， a_2 代表题目难度， a_3 代表能力层次， a_4 代表题目所考察的知识点、 a_5 代表题目的区分度， a_6 代表题型。那么，组成一份总共有 m 道试题的试卷，如果每道试题有 n 项属性，就相当于构建了一个 $m \times n$ 的目标矩阵 S 。

$$S = \begin{bmatrix} a_{11} & a_{12} & \cdots & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & \cdots & a_{mn} \end{bmatrix} \quad (5.1)$$

目标矩阵 S 其实是一个问题求解的目标状态矩阵，且目标状态不是惟一的。目标矩阵应满足以下的约束条件^[31]：

(1) 试卷总分为 Z ：

$$Z = \sum_{i=1}^m a_{i1} \quad (5.2)$$

(2) 试卷难度系数为 D ：

$$D = \sum_{i=1}^m a_{i1} a_{i2} / \sum_{i=1}^m a_{i1} \quad (5.3)$$

(3) Z_p 为第 p 教学要求的分数，教学要求（了解、理解、掌握、应用）和所占分数由用户给定，即教学要求约束。其中 $c_{3i} = \begin{cases} 1 & a_{i3} = p \\ 0 & a_{i3} \neq p \end{cases}$ ；

$$Z_p = \sum_{i=1}^m c_{3i} a_{i3} \quad (5.4)$$

(4) 试卷区分度为 Q ：

$$Q = \left(\sum_{j=1}^m a_{j1} a_{j5} \right) / \sum_{i=1}^m a_{i1} \quad (5.5)$$

(5) Z_q 为第 q 种题型的分数，其中 $c_{6i} = \begin{cases} 1 & a_{i6} = q \\ 0 & a_{i6} \neq q \end{cases}$ ；

$$Z_q = \sum_{i=1}^m c_{6i} a_{i6} \quad (5.6)$$

(6) Z_h 为第 h 知识点的分数，其中 $c_{4i} = \begin{cases} 1 & a_{i4} = h \\ 0 & a_{i4} \neq h \end{cases}$ ；

$$Z_h = \sum_{i=1}^m c_{4i} a_{i4} \quad (5.7)$$

5.2.2 组卷算法的设计

在考试系统中，采用了基于遗传算法的组卷算法。遗传算法的几个主要特点，如直接对结构对象进行抽象，不存在求导和函数连续性的限定；具有内在的隐并行性和更好的全局寻优能力；采用导向式概率化的寻优方法，能自动获取和指导优化的搜索空间；自适应地高速搜索方向，并且不需要确定规则等都适宜于处理自动组卷的问题。

但是传统的遗传算法首先生成一定规模的初始群体，然后使其中的个体以一定的概率进行交叉与变异，实现个体结构的重组，再按预定的评价函数选择复制优秀个体，组

成新的一代，如此循环迭代，以期最终找到满足寻优条件的全局最优解。但是，这样的算法存在着越道搜索后期效率越低下，并且容易产生未成熟即收敛的情况。

针对系统自动组卷的具体情况，本文主要从适应度函数、初始种群、控制参数等几个方面对遗传算法加以改进，使其能够很好满足各项组卷需求。

(1) 初始种群

初始种群的特性对遗传算法的计算结果和计算效率均有重要影响，算法要实现全局最优，初始种群在解空间中应该尽量分散。而在传统遗传算法中，初始种群是随机产生的，所以，为了加快遗传算法的收敛并减少迭代次数，初始种群的生成要满足题型、题量和试卷总分的要求，这样能够有效提高求解速度。

(2) 适应度函数

在遗传算法中，采用适应度函数值是来评判群体中的个体优劣，一般情况下，适应度函数值越大的个体越好，即表示这个试题的组合的各项约束条件越接近用户指定的理想值。适应度函数值是遗传算法进行优化所用的主要信息，它与个体的目标值存在一种对应关系。遗传算法利用适应度值这一信息来指导搜索方向，根据约束条件，建立目标函数为误差函数，另外，根据实际组卷经验，对不同的约束条件可给定不同的允许误差（0.01~0.05），只要试卷个体满足第*i*项组卷要求的误差在容差范围内，即可认为第*i*项组卷要求的误差为0，这样以加快搜索到合理解的速度，由目标函数来设计适应度函数，而不需要适应度函数连续或可导以及其它辅助信息。我们采用以下形式的适应度函数：

$$F = \frac{1}{\left(1 + \sum_{i=1}^n k_i |e_i|\right)} \quad (5.8)$$

其中 $e_i (0 \leq e_i \leq 1)$ 对应为第*i*项组卷因素对组卷约束程度的归一化相对误差， $k_i (0 \leq k_i \leq 1)$ 为的相应的误差权值系数，适应度函数可以较好的反映求解组卷问题的特征，当试卷个体对各项组卷约束条件的相对误差越小时，它的适应度函数值就越大，则表示试卷个体越接近组卷目标。

(3) 遗传算子

① 选择算子：选择操作指从群体中按个体的适应度函数值选择出较适应环境的个体。选择将使适应度高的个体繁殖下一代的数目较多，而适应度较小的个体，繁殖下一代的数目较少，甚至被淘汰。本文采用期望值模型选择机制，首先计算出群体中所有个

体期望被选中的次数 $N_i = \frac{M \cdot F_i}{\sum_{i=1}^M F_i}$ (M 为群体规模, F_i 为第 i 个个体的适应值), 然

后根据 N_i 的整数部分确定个体 i 被选中的次数。然后对 N_i 的小数部分作为概率进行贝努利试验, 如果试验成功, 则该个体被选中。这样, 不但个体适应值高的个体更容易被选中, 而且即便是适应值低的个体也更有可能会被选中。

② 交叉算子: 在遗传算法中, 交叉算子将被选中的两个个体的基因按一定的交叉概率, 即 P_c 进行交叉, 从而生成两个新的个体。这里将以上选出的个体进行两两随机配对, 对每一对相互配对的个体采用有条件的“均匀交叉”, 即两个配对个体的每一个基因座上的基因都按设定的交叉概率 P_c 和一定的条件(确保交换后个体仍是有意义的组合)进行交换, 并产生两个新个体。

③ 变异算子: 变异算子的基本内容是对群体中的个体串的某些基因位置上的基因值作变动。对变异算子的改进主要是在同一题型段内进行有条件的单点变异, 并且变异只针对交叉后的个体。

(4) 控制参数

交叉是优化新生中的一个重要步骤, 在传统遗传算法中, 交叉概率 P_c 是个常数, 但在实际情况中, 优良的交叉率与遗传代数间的关系较大。在迭代初期, P_c 如果选的大, 可以造成足够的扰动, 从而增强遗传算法的搜索能力; 在迭代后期, P_c 选的小, 可以避免破坏优良基因, 加快收敛速度。同理, 变异是优化新生中的另一重要步骤, 如果变异概率 P_m 是常数, 则群体的性质会趋于一致。因此, 无论是交叉概率, 还是变异概率都要随着进化的进行而不断调整。

P_c 和 P_m 的选取直接影响算法的收敛性。 P_c 越大, 新个体产生的速度越快, 而遗传模式被破坏的可能性也越大, 这样高适应度的个体结构也可能被破坏; 而 P_c 过小, 会使搜索过程变慢。另一方面, 如果 P_m 取值过大, 遗传算法就则成为了纯粹的随机搜索算法; 而 P_m 过小, 则不容易产生新的个体结构, 使算法早熟, 陷入局部最优。所以, 为了加快遗传算法的搜索效率, 并有效地防止其陷入局部最优, 保持优良试卷个体, 在本论文中根据种群的进化情况采用了自适应函数来动态地调整交叉概率 P_c 和变异概率 P_m , 交叉率和变异率随着个体的适应度在种群平均适应度和最大适应度之间进行线性调性。公式如下所示:

$$P_c = \begin{cases} 0.9(f_{\max} - f)/f_{\max} & f \geq f_{\text{avg}} \\ 0.9 & f < f_{\text{avg}} \end{cases} \quad (5.9)$$

$$P_m = \begin{cases} 0.3(f_{\max} - f')/f_{\max} & f' \geq f_{\text{avg}} \\ 0.3 & f' < f_{\text{avg}} \end{cases} \quad (5.10)$$

其中， f_{\max} 为群体最大适应度值， f_{avg} 为群体平均适应度值， f 为两个交叉个体中的适应度值较大的一个， f' 变异个体的适应度值。

从公式可以看出，对于高于平均适应度值的个体，其受破坏的可能性较小，而对于低于平均适应函数值的个体，其受破坏的可能性较大，从而被淘汰掉。因此，自适应的 P_c 和 P_m 能够为每个试卷个体提供一个最佳的 P_c 和 P_m 。这种自适应遗传算法在保持试卷群体多样性的同时，保证遗传算法的收敛性并防止遗传算法陷入局部最优。

(5) 最优个体保存策略

最优个体保存策略的改进是将当前解群体中的适应度最高的个体结构完整的复制到下一代种群中。这样主要是为了使进化过程中某一代的最优解可以不被交叉或变异操作所破坏。在算法实现时，让适应度最高的个体直接进入下一代中，并将其设为种群中的第一个个体，其他不足种群数量的个体继续进行选择，直到达到原始种群的数目。这种方法既可以保证优良品种的个体得到更多的繁殖机会，又可以防止少数几个优良品种由于过度繁殖而控制整个种群，使得算法陷入到局部最小。

(6) 迭代终止条件

在改进的遗传算法中，设置了期望适应度值，再设置最大迭代次数，把每一代计算出来的最大适应度个体的适应度值与期望适应度值相比较，当适应度最高的个体的适应度值比期望适应度值大时，算法会停止迭代，否则继续进行遗传操作、计算适应度值、反复迭代直到适应度值达到期望值或达到最大迭代次数则停止迭代，如达到期望适应度值组卷成功，否则组卷失败。

5.3 组卷算法的实现与分析

5.3.1 组卷算法的流程

根据系统的具体需要，我们将上述组卷策略和改进的遗传算法应用于组卷系统中，并在具体的应用环境中对环境变量进行改造，使得组卷算法更加高效。这种改进的自动组卷算法的具体算法流程图如图 5.1 所示。

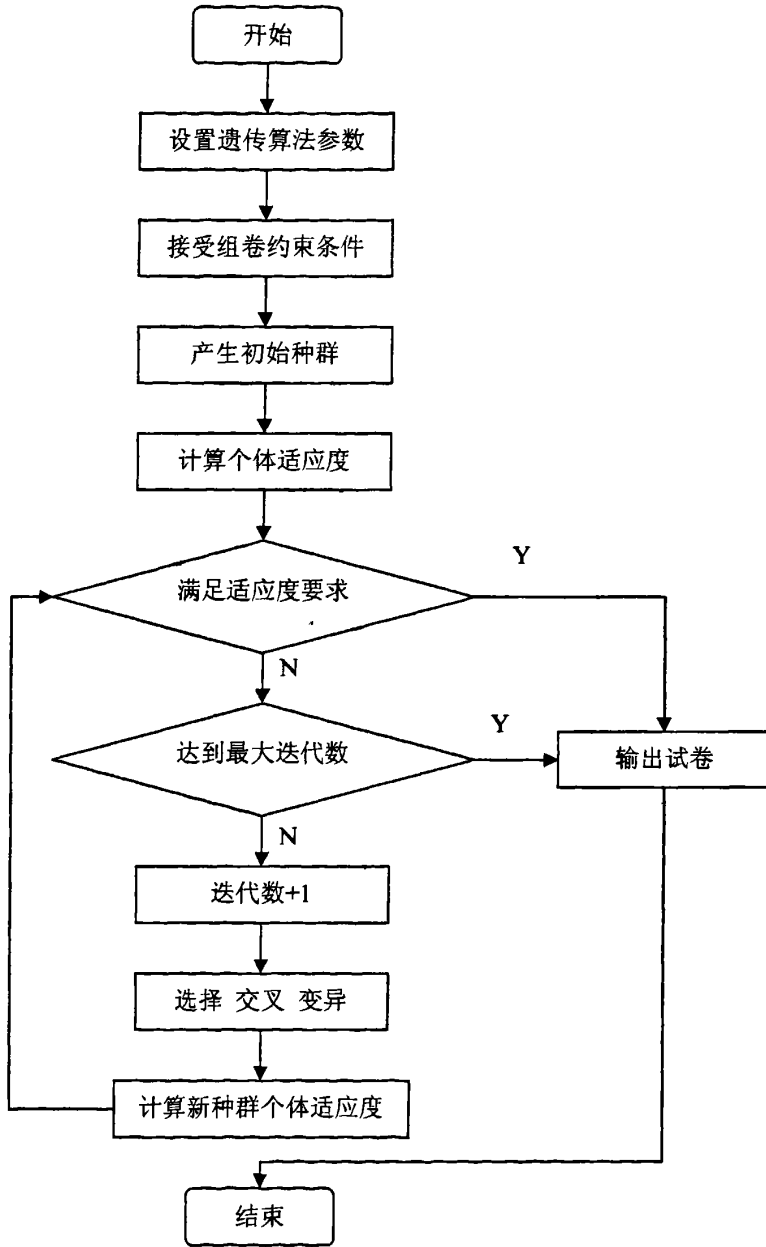


图 5.1 自动组卷算法流程图

Fig. 5.1 The of algorithm of auto generating paper flow chart

5.3.2 组卷算法的实验分析

建立测试题库对该组卷算法进行测试，题库中共包含 850 道题目，分为四种题型，六项知识点，按照表 5.1 所示的试卷约束条件，进行了组卷测试。

其中，试卷中各题型的分值及分布如表 5.2 所示，试卷中各知识点的分值及分布如表 5.3 所示。

表 5.1 试卷的约束条件
Tab. 5.1 Constraint conditions of test paper

考试时间	总分	区分度	难度
100 分钟	100 分	0.4	0.65

表 5.2 试卷中各题型的分值及分布
Tab. 5.2 Every type of questions marks and distribution of test paper

题型	题型 1	题型 2	题型 3	题型 4
题目数目	20	15	4	2
每题分值	2	2	5	5

表 5.3 试卷中各知识点的分值及分布
Tab. 5.3 knowledge points marks and distribution of test paper

知识点	知识点 1	知识点 2	知识点 3	知识点 4	知识点 5	知识点 6
分数	20	20	20	20	10	10

固定最大代数为 300，群体规模为 50，算法终止条件是迭代数达到最大代数或最好个体的目标函数值，测试交叉概率 P_c 和变异概率 P_m 对程序收敛性的影响。

(1) 当交叉概率 $P_c = 0$ ，变异概率 $P_m = 0$ 时，随机运行程序 10 次，组卷的成功率和效率很低。因为当 P_c 和 P_m 均为 0 时，算法过程中只有选择操作而无交叉、变异操作，无法通过交叉、变异操作产生新的更优的个体，只能简单地复制每一代的优胜个体到下一代中，因此搜索空间很小，很难找到全局最优或较优解。

(2) $P_m = 0.001$ ，对 P_c 分别取不同值进行 12 次组卷实验，实验结果如表 5.4 所示。可见， P_c 在 0.2~1 之间取值均可，但组卷成功次数和平均迭代数随 P_c 的增大呈现先增后降的趋势。由于交叉条件的约束， $P_c = 1$ 时并不是所有的个体都进行交叉操作，故此时组卷效果也较好。

表 5.4 参数 P_c 对组卷的影响Tab. 5.4 Influence of parameter P_c to generating paper

P_c	组卷成功次数	平均迭代数
0	7	240.2
0.2	9	153.3
0.4	10	140.9
0.6	11	138.2
0.8	10	146.3
1.0	8	150.3

(3) $P_c = 0.6$, 对 P_m 分别取不同的值进行 12 次组卷实验, 结果如表 5.5 所示, 可见 P_m 大于 0.05 时, 组卷效率明显降低, 这是因为变异虽然增加了群体的多样性, 但是过多的变异会破坏群体的优良模式, 甚至会使遗传算法变成随机搜索。若取值过小, 产生新个体的速度又太慢。

表 5.5 参数 P_m 对组卷的影响Tab. 5.5 Influence of parameter P_m to generating paper

P_m	组卷成功次数	平均迭代数
0.001	9	148.4
0.01	12	136.7
0.05	8	180.1
0.1	4	253.6

固定 $P_c = 0.6$, $P_m = 0.05$, 对群体规模分别取不同的值进行 12 次组卷实验, 测试群体规模对程序收敛性的影响, 测试结果如表 5.6 所示。

表 5.6 群体规模对组卷的影响

Tab. 5.6 Influence of population size to generating paper

种群规模	组卷成功次数	平均迭代数
20	6	250.9
40	12	142.6
80	12	152.6
100	12	178.2
200	12	190.3

可以看出,当群体规模较小时,获得有效试卷的成功率较小,这是由于群体规模较小时,群体中的个体缺乏多样性,算法的搜索空间分布范围有限,因而搜索可能停止在未成熟阶段,引起未成熟收敛现象。随着群体规模的增大,算法的搜索空间增大,获得最优解的机会提高,生成满足要求试卷所需的迭代数减少。但这并不能说明群体规模越大越好。群体规模太大,会使得搜索空间很大,降低程序的运行效率。在本算法中,群体规模在 40~80 之间取值,效果最好。

在一个能够进行自动组卷的考试系统中,组卷性能的高低主要取决于组卷算法是否高效,而传统的组卷算法具有成功率较低,而且组卷条件不能太复杂。并且组卷所需的时间和所耗费的系统资源也较大。通过改进遗传算法,并将其应用到组卷系统的组卷算法中,能够使组卷的成功率和收敛速度都得到明显的提高,这是由于改进的遗传算法实行了全局并行搜索,搜索空间大,并且在搜索过程中不断向可能包含最优解的方向调整搜索空间,从而易于找到最优解,同时也能够满足更加复杂的组卷条件。

实验结果表明,在进化到 140 代左右时即可生成一份满意的试卷且误差比较小,改进的遗传算法能有效地解决智能组卷问题。由于求解精度和收敛速度是相互矛盾的;要使组卷的误差更小和收敛速度更快,还需更进一步地深入的研究。

6 考试系统的实现

6.1 用户登录

登录是用户进入考试系统并进行其他操作的第一步，系统中的所有合法用户，都必须在登录页面输入的用户名及密码，通过验证后才能够进入系统，并根据用户类型获取相应的操作权限。

在系统的登录界面中，首先要对用户输入的登陆信息进行合法性验证，然后再通过与数据库中的正确信息核对已确定用户身份。如通过验证，则由登录页面转到相应用户的主页面，例如，用户成功登陆后，考生可以进入在线考试、成绩查询等前台页面，而教师则能够进入题库管理，试卷管理等后台页面。如果未通过验证则提示输入错误，返回登录界面。在用户登录系统期间，系统将使用 Session 对用户权限进行管理。用户登录的部分代码如下：

```
public class User{
    .....
    //获取用户数据
    public static User getUser(String username, String password){
        User user=new User();
        Statement stmt=conn.createStatement();
        //建立 SQL 查询语句
        String query ="SELECT count(*) FROM user_in this_system where user
account_id="+request.getParameter("username")+"" and password="+digestResult +""";
        //从数据库中查询出相应的用户数据
        ResultSet result=stmt.executeQuery(query);
        If (result.next()){
            user.setUsername(result.getString("username"));
            user.setPassword(result.getString("password"));
            return user;
        }
    }
}

public class logAction{    //验证登录信息
    public actionForward perform(ActionMapping mapping, ActionForm
HttpServletRequest request, HttpServletResponse respinse){
```



```

DynaValidatorForm userForm=(DynaValidatorForm) form;
//从客户端获取 Session 信息
HttpSession session=request.getSession();
String username=(String)userForm. get("username");
String password=(String)userForm.get("password");
//从数据库获取用户信息
User user=new User();
User.getUser(username, password);
session.setAttribute("user", user_);
..... //跳转到相应页面
}
}
    
```

6.2 题库管理

试题库的管理主要包括题目的录入、修改和删除，而为了建立试题库，首先需要教师向题库中添加考试题目。教师在登录到考试系统后，可以选择进入题库管理页面，在此页面上，教师不但能够完成向题库中添加试题，还能够完成题目的修改和删除等其他管理操作。

查询试题			添加试题			批量删除		
<input type="checkbox"/>	题型	序号	科目	章节	题目	分值	操作	
<input type="checkbox"/>	单选题	151	计算机文化基础	3	在EXCEL中，A5的内容是“A5”，拖动填充柄至C5，则B5、C5单元格的	1	修改	删除
<input type="checkbox"/>	单选题	152	计算机文化基础	3	在EXCEL的单元格中，如果输入数字字符串068812344（电话号码）的	1	修改	删除
<input type="checkbox"/>	单选题	153	计算机文化基础	3	在建立EXCEL工作表时，哪项操作可以删除工作表第4行。	1	修改	删除
<input type="checkbox"/>	单选题	154	计算机文化基础	3	以下数据中，不能输入到EXCEL工作表的单元格中的是：	1	修改	删除
<input type="checkbox"/>	单选题	155	计算机文化基础	3	在EXCEL中，一个新工作簿默认有多少个工作表。	1	修改	删除
<input type="checkbox"/>	单选题	158	计算机文化基础	3	单元区域（B14、C17、A16、D18、C15、E16）包括的单元格数目是：	1	修改	删除
<input type="checkbox"/>	单选题	157	计算机文化基础	3	在EXCEL中，创建公式的操作步骤有：1.在编辑栏键入“=”；2.键入	1	修改	删除
<input type="checkbox"/>	单选题	158	计算机文化基础	3	在EXCEL的数据清单中，利用自动筛选可对几列的数据筛选出符合条件	1	修改	删除
<input type="checkbox"/>	单选题	159	计算机文化基础	3	打印EXCEL工作表时，欲使每页都打印顶端标题行，应在页面设置对话	1	修改	删除
<input type="checkbox"/>	单选题	160	计算机文化基础	3	在EXCEL中，可以一瞥显示多行文本的单元格的格式设置是：	1	修改	删除

图 6.1 题目管理界面

Fig. 6.1 Questions management interface

6.2.1 题目录入

首先介绍题目录入，在向题库中录入试题之前，除了要准备好合乎格式试题的题干和标准答案，还要为每道题目设定题号、分值、难度、区分度等与系统自动组卷有关的其他题目属性。

科目	计算机文化基础	▼	分值	2
题型	多选题	▼	难度	请选择
章节	请选择	▼	区分度	请选择
知识点	请选择	▼	掌握程度	请选择
<input type="text"/> <input type="text"/>				
正确选项个数	4	▼	正确答案	A <input type="checkbox"/> B <input type="checkbox"/> C <input type="checkbox"/> D <input type="checkbox"/>
A:	<input type="text"/>			
B:	<input type="text"/>			
C:	<input type="text"/>			
D:	<input type="text"/>			

图 6.2 题目录入界面

Fig. 6.2 Questions add interface

在开始录入试题的时候，首先要依次输入题目的属性和内容，根据题型的不同，系统会调用不同的试题录入页面，之后系统会对之前输入进行核对，如果出错，则提示重新输入；如果输入无误，则在数据库中建立相关的题目记录，即将题目保存到是题库中。试题录入的活动图如图 6.1 所示。其中，实现题目录入功能的部分代码如下所示。

```
function saveQuestion(){
    ..... //获取题型、题目内容和答案等信息
    if (sContent==""){
        alert("请输入题目内容!");
        return;
    }
    if(sKey==" " && !confirm("您没有输入题目答案，仍然继续保存吗?")){
        return;
    }
}
```

```

}
form1.txtContent.value=sContent; //记录题目内容
form1.txtKey.value=sKey; //记录答案
..... //向数据库提交更改
}
    
```

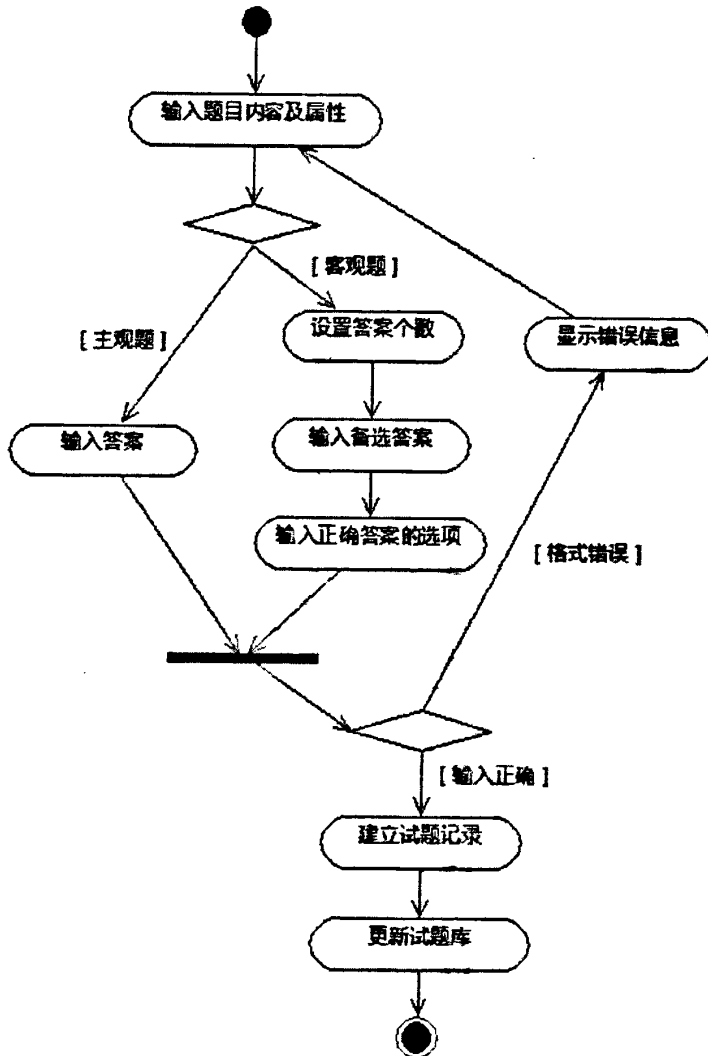


图 6.3 题目录入活动图

Fig. 6.3 The activity diagram of increasing questions

6.2.2 题目修改与删除

教师对试题库的管理,除了能够进行题目录入,还能够对题目进行修改与删除操作。如果选择对题目进行修改,系统则将需修改的题目直接显示在题目录入页面上以供修改,之后的操作与录入题目一致。如果选择删除题目,系统则根据所选题目编号直接从试题库中将题目删除,并更新数据库。

6.2.3 题库检索

试题库中的题目数目庞大,所以,当教师或管理员浏览器库的时候,系统应该提供完善的题目检索功能。在查询题库前,首先输入查询条件,例如题型、题号、难度等,然后系统会从试题库中返回所有符合条件的题目,以供下一步操作。如果没有符合条件的题目,则返回错误提示。对题库检索的相关步骤进行描述的活动图如图 6.2 所示。

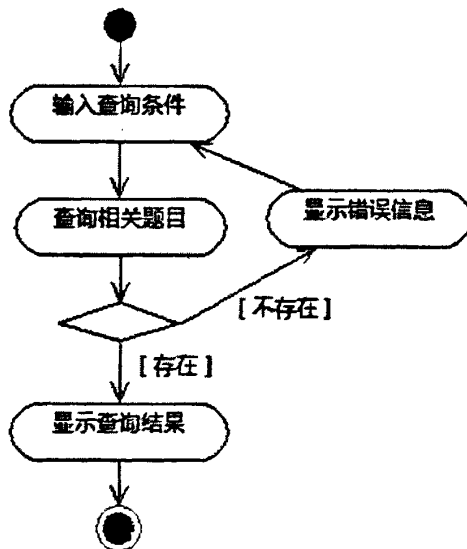


图 6.4 题库检索活动图

Fig. 6.4 The activity diagram of searching question database

6.3 考试管理

考试管理的主要功能是指在考生进行在线考试期间,管理员对各考场中的考生进行实时监控。考试系统在能够在事先安排好的时间段内向考生开放考试。开考后,管理员可以查询具体所有已登录系统的考生的个人信息及考试状态,如是否已成功抽取试题,

是否交卷，另外，管理员还可以即时查询考场情况，包括本次考试的应考人数，实考人数，已交卷人数，为交卷人数等信息。

序号	考试名称	考场	考试时间	考生人数	状态	操作
1	计算机文化基础	C102	2008-08-02 08:00:00—2008-08-02 09:30:00	59	正在进行	查看详情信息
2	计算机文化基础	C103	2008-08-02 08:00:00—2008-08-02 09:30:00	70	正在进行	查看详情信息
3	计算机文化基础	C201	2008-08-02 08:00:00—2008-08-02 09:30:00	65	正在进行	查看详情信息
4	计算机文化基础	C201	2008-08-02 10:00:00—2008-08-02 11:30:00	70	等待开始	查看详情信息
5	多媒体硬件制作	B303	2008-08-02 09:00:00—2008-08-02 10:10:00	50	等待开始	查看详情信息
6	多媒体硬件制作	B304	2008-08-02 09:00:00—2008-08-02 10:10:00	48	等待开始	查看详情信息

图 6.5 考试管理界面

Fig. 6.5 Exam management interface

考试结束后，系统会自动结束考试状态并收取所有尚未提交的试卷。同时，如果遇到某些其他状况，管理员也可手动调整考试的时间安排，以提前结束或者延长考试时间。而系统计算考试剩余时间是在客户端执行的，倒计时功能采用 JavaScript 进行实现，主要利用了 Windows 对象的 setTimeout (expression, msec)方法，该方法能够每隔 msec 毫秒，重新执行表达式 expression。程序的部分代码如下：

```

var intMinutes=document.form1.hiddenMinute.value;
var intSeconds=document.form1.hiddenSecond.value;
function tick(){ //用于考试计时的函数
    var minutes, seconds;
    if (intSeconds=0){
        intMinutes=intMinutes-1; intSeconds=60;}
    intSeconds=intSeconds-1;
    if (intMinutes<10){
        minutes="0"+intMinutes+"minutes ";}else{
        minutes=intMinutes+"minutes";
    }
    if (intSeconds<10){
        seconds="0"+intSeconds+"second";
    }else{
        seconds=intSeconds+"second";
    }
    timeString=minutes+seconds; //考试当前剩余的时间

```

```

        window.setTimeout("tick();", 1000);    //每过 1 秒，需重新计算
    }
    window.onload=tick;
}

```

需要注意的一点是，当考试结束前后，考试系统中会出现交卷的瞬时高峰，在短时间内，大量的考试几乎同时在向服务器提交试卷，此时极易造成阻塞，甚至瘫痪服务器。这样的现象主要因为在考试结束时，服务器端的远程模块连接数过多，占用了大量的系统资源。所以，为了避免这种情况，在交卷的时候，系统先对所有的交卷考生进行分组，然后按组依次交卷。

6.4 在线考试

考生在开始考试之前，首先要登录考试系统，在考生登录时，首先要对考生的用户名和密码进行验证，如果有误，则提示考生重新输入。另外，即便考生成功登陆，还要将考生的个人信息对考生进行提示。核对无误后，考生首先要选择本次考试的科目，在得到由系统自动抽取的试卷后，开始答卷，答卷期间，考生可以自行交卷，如果知道考试规定时间结束还未交卷，系统会自动提交试卷，并结束当次考试。在线考试的活动图如图 6.3 所示。

一、单选题（每题1分）

1、在WINDOWS XP中，用户可以对多个窗口进行层叠式或平铺式排列，下面操作正确的是：

- A. 用鼠标右键单击“桌面”空白处，然后在弹出的快捷菜单中选取要排列的方式。
- B. 用鼠标右键单击“任务栏”空白处，然后在弹出的快捷菜单中选取要排列的方式。
- C. 在“资源管理器”窗口中的“查看”菜单下选择“排列图标”。
- D. 在“我的电脑”窗口中的“查看”菜单下选择“排列图标”。

2、以下说法中，正确的是：

- A. 同一文件夹中允许有两个文件的文件名相同，但它们的内容必须相同。
- B. 不同文件夹中不允许有文件名相同的文件。
- C. 不同文件夹中，允许有文件名相同的文件，但它们的内容必须相同。
- D. 不同文件夹中，允许有文件名相同的文件。

图 6.6 考试界面

Fig. 6.6 Examination interface

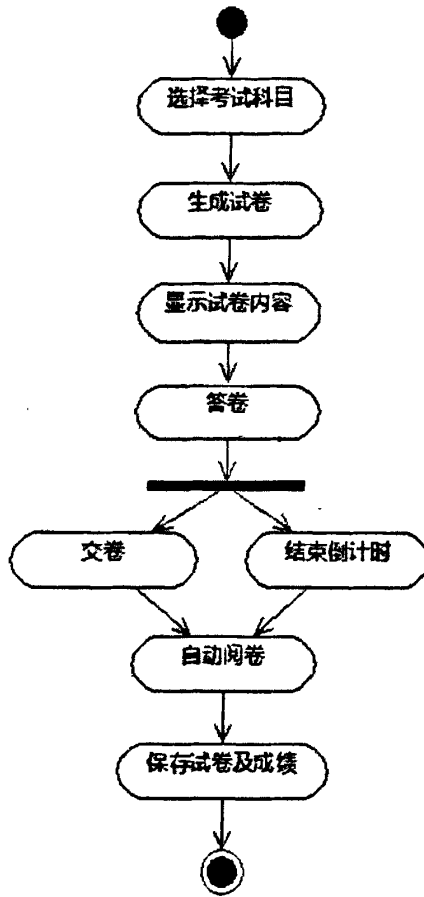


图 6.7 在线考试活动图

Fig. 6.7 The activity diagram of test

考生通过网络登录考试系统在线答卷时，由于各种各样的客观原因，断线或是答题时死机的情况很难避免，而当考生再次登录考试系统的时候，如果再次重新抽题组卷，势必影响到考试的公平和考生的成绩。所以，在考生答卷的同时，对试卷和考生答案等临时数据进行随时保存就显得十分必要。可是，如果所有考试客户端都频繁的想考试服务器保存数据，势必要加重服务器和网络的负担。的因此，考试系统默认为每隔三分钟自动将考生的试卷及答案保存到服务器中。但是，在考生断线后重新登陆系统后，并不需要重新抽题组卷。为了解决这个问题，系统还设计了考生重新登录功能。

重新登陆的设计原理是利用考生用户的 Session 实现的。当考生成功登陆之后，会产生一个为该考生专用的 Session。这个 Session 可以在没有连接的情况下依然拥有三十

分钟的生存周期。那么，如果在半小时内，服务器仍然没有收到考试客户端发送的数据或者消息时，Session 才会失效。而当考生在断线后三十分钟内重新登录系统的时候，Session 则将之前保存在服务器中的临时数据重新传输到考生客户端，使考生能够依然使用断线前的试卷并继续答卷。部分代码如下：

```
class examTempData {
    private int examTimeCur=0; //考试已经过的时间。
    private table examTemp;    //临时表，用于保存考生的试卷和答案
    private int timer=0;      //断线计时器
    public void LoginBreak(Session session)
    {
        if(GetTempValue(session)==true){ //能够正确读取临时数据
            timer=0;
        }
        else{ //如果超过半小时，则关闭 Session
            if (timer>=1800){
                .....
                session.invalidate();
            }
        }
    }
    public bool SndLogin(Session session)
    //检查用户是否断线过，如果未曾断线，则重新登陆
    if (session.getAttribute(name)=NUL)
        return false;
    else{
        log.Login(session);
        timer=0;
    }
    return true;
}
```


结 论

本考试系统经过对 JSP 技术和 JavaBeans 组件技术进行了深入研究,采用 JSP 和 JavaBeans 开发动态页面,系统采用了 B/S 三层体系架构模式,各层之间相对独立,使系统的扩充性好、安全性高,代码易维护和可重用性好。此外,考试系统采用了改进遗传算法,使得系统自动组卷速度快、成功率高、效果好;并且具有一定的通用性,可以用于多种课程的考试;同时,又具有较完善的辅助功能,比如客观题自动评分功能,良好的考试现场恢复机制等。

本课题的开发,采用了面向对象的设计建立系统的分析模型和设计模型,并参考了其他一些先进的网络考试系统的设计,吸收了国内外相关领域的研究成果,从而使本系统在理论上和实用上都有所突破。投入使用后,能够明显改善教学环境,减轻教师工作强度,提高教学效率和考试评估的科学性、准确性。

由于时间和试验条件以及经验等诸多限制,完成系统开发之后,仍有一些不足之处有待改进。下一步的开发工作主要有:

- (1) 增加题库中的题型数量,并且能够实现多媒体信息嵌入的支持。
- (2) 进一步完善成绩统计分析功能。要能通过分析考生历次考试成绩及试卷中各题得分,了解其进步情况、知识要点掌握情况,为教学提供更大帮助。
- (3) 当前的考试系统仅仅能够对客观题进行自动批阅,所以,应拓展自动阅卷功能,使其能够准确的进行主观题的自动批阅。

以上不足之处今后将通过今后的学习和实践,不断的进行补充和完善,使考试系统能满足日益扩展的考试需求,在实际应用中发挥更大的作用。

参考文献

- [1] 杨春明, 廖竞. Web 方式下通用考试系统的设计与实现. 计算机与现代化. 2004, 20(1):11-13.
- [2] 唐章蔚. 现代远程教育综合管理系统设计与实现. 微计算机信息. 2006, 22(21):34-36.
- [3] 侯鸿林, 朱向峰. 基于 WEB 的考试系统的实现. 教育信息化. 2004, 10(1)39-41.
- [4] 殷大发. 无纸化考试系统的研究和开发. 计算机与现代化. 2004, 20(3):82-85.
- [5] Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide, 北京:机械工业出版社, 2001.
- [6] Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language Reference Manual, 北京:机械工业出版社, 2001.
- [7] Wendy Boggs, Michael Boggs. Mastering UML with Rational Rose 2002. New York: Sybex Books, 2002.
- [8] Craig Larman. Applying UML and Patterns Training Course: A Desktop Seminar from Craig Larman. New York: Prentice Hall PTR, 2002.
- [9] 张龙洋. UML 与系统分析设计. 北京:人民邮电出版社, 2001.
- [10] Scott W. Ambler. The Application Developer's Guide to Object Orientation and the UML. 北京:机械工业出版社, 2003.
- [11] James Rumbaugh, Ivar Jacobson, Grady Booth. The Unified Software Development Process. 北京:机械工业出版社, 2002.
- [12] 吴际, 金茂忠. UML 面向对象分析. 北京:北京航空航天大学出版社, 2002.
- [13] Reese, George. Database Programming with JDBC and Java. New York: O'Reilly Media, Inc, 2000.
- [14] 周慧贞, 刘晓华, 张健. JSP 应用开发详解(第二版), 北京:电子工业出版社, 2004.
- [15] 林邦杰. JSP 交互网站实务经典. 北京:中国青年出版社, 2001.
- [16] Andrew Harbourne Thomas, Sam Dalton, Simon Brown et al. Professional java Servlets 2.3. New York: Peer Information, 2002.
- [17] Karl Avedal. JSP 的编程指南. 黎文译. 北京:电子工业出版社, 2001.
- [18] 张白一, 崔尚森. 面向对象程序设计——Java. 西安:西安电子科技大学出版社. 2003.
- [19] Tom Myers, Alexander Nakhimovsky. Professional Java XML Programming with Servlets and JSP, New York: Peer Information Inc, 1999.
- [20] Mark Ashnault, Ziyad Dean, Thomas Garben et al. J2EE 技术内幕. 刘堃译. 北京:机械工业出版社, 2002.
- [21] Subrahmanyam Allamaraju, Andrew Longshaw, Daniel O'Connor et al. J2EE 编程指南. 马树奇译. 北京:电子工业出版社. 2002.
- [22] 赖碧云, 李小丹, 章少强. 网络数据库系统开发中 JDBC 的应用. 现代计算机. 2003, 20(1):76-79.
- [23] 张明辉. 基于 JDBC 技术的数据库连接和访问的研究. 电脑与电信. 2008, 14(6):49-51.

- [24] 范立南. SQL Server 2000 实用教程. 北京:清华大学出版社, 2004.
- [25] 陈浩奎. SQL Server 2000 存储过程与 XML 编程. 北京:清华大学出版社, 2003.
- [26] 安吉尧, 喻飞, 李仁发, 李蕊. 电子考试中动态抽题算法的应用研究与实现. 计算机工程与设计. 2004, 25(2):188-190.
- [27] 殷凤琴, 秦奋涛. 遗传算法算子优化的应用. 现代计算机, 2008, 25(8):21-23.
- [28] 王小平, 曹立明. 遗传算法——理论、应用与软件实现. 西安:西安交通大学出版社, 2002.
- [29] 刘彬, 李勇, 糜长军. 智能组卷系统中专家知识的表示与实现. 计算机工程与应用, 2002, 39(17):299-231.
- [30] 孟朝霞. 基于自适应免疫遗传算法的智能组卷. 计算机工程, 2008, 34(14):203-205.
- [31] 段朋鹏, 郭庆平. 改进遗传算法在智能组卷系统中的应用. 软件导刊, 2008, 7(3):118-120.

致 谢

在本论文完成过程中，得到许多老师、同学和朋友们的鼓励帮助和关怀，在此谨向所有帮助和关心过我的人们表示衷心的感谢。

首先要感谢我的导师李明楚教授，在课题研究和论文撰写的过程中都得到了李老师的精心指导和帮助。他扎实的理论基础、踏实的工作作风和严谨的治学态度使我颇为受益。

此外，还要衷心感谢大连理工大学软件学院的各位老师，在这几年的学习时间里，是你们的帮助和关心，是我在学业上能够有所进步，并在此论文的完成过程中给予了不断的帮助。

最后还要特别感谢我的家人和许多同学，他们在学习和生活上给予我无私的帮助，使我在科研的道路上无所羁绊；不断给我以激励与鼓舞，使我拥有迎接和战胜一切困难的勇气和决心。

基于遗传算法的考试系统的设计与实现

作者: [孟禹](#)
 学位授予单位: [大连理工大学](#)

相似文献(10条)

1. 学位论文 [宋晓俊](#) [基于J2EE架构的在线考试系统研究与开发](#) 2008

计算机网络和多媒体技术在现代社会的各个领域中被广泛应用,带来了计算机远程教育的飞速发展。然而,随着教学内容的不断丰富和参加远程教育人数的日益增多,对远程教育系统的要求也越来越高。在线考试系统作为远程教育系统中的—一个重要环节,对远程教育系统的优劣有重要的影响。所以,研究和开发一个具有强大功能,且可扩充、易维护的在线考试系统有着十分重要的现实意义。

本文设计并实现了一个基于Struts框架的在线考试系统,可以为学生提供在线考试服务。开发该系统的目的是通过网络实现试题资源的共享和快速分发,为参加远程教育的学生提供在线测试功能。

该系统采用了基于MVC设计模式的Struts框架的三层结构。在这三层结构中,视图层是考生看到并与之交互的界面,模型层负责业务流程、状态的处理以及业务规则的制定,控制层是从用户接收请求,将模型与视图匹配在一起。Struts三层框架结构可以使得该系统结构更加清晰,而且使该系统更具有良好的可扩展性、可维护性以及平台无关性。

为了能够实现组卷系统的智能化,本文在该系统的自动组卷模块中使用了遗传算法,并对经典遗传算法进行了一些改进。在遗传算法的编码方式中采用分组自然数编码方式代替二进制编码方式,这样减少了染色体长度空间,编码直接采用试题编号,省去了编码和解码的繁琐。并且在算子选择中,我们采用了模拟小生境的方法,它能够有效地维持种群的多样性,从而避免产生局部最优解,改善未成熟收敛。运用自适应理论改进了交叉概率和变异概率,使算法本身总能找到适合自己的交叉概率和变异概率,提高了系统的收敛速度,同时也避免了遗传过早进入“局部收敛”状态。

2. 学位论文 [尹志东](#) [在线考试系统研究与实现](#) 2009

随着教育信息化的发展,在线考试系统作为一种新型的考核手段和考试方式,在实际教学中得到了广泛的应用,并成为当前计算机信息系统研究与应用的热点领域。本考试系统在综合分析了当前考试系统研究成果的基础上,采用B/S和C/S两种结构相结合的模式开发,利用IE浏览器作为界面设计,使用户可利用浏览器直接访问本平台,完成考试和测评。NET技术和SQL数据库以及其他网络程序设计开发平台为本系统主要开发技术,本学位论文对在考试系统的最重要功能,包括用户登录、身份验证、新用户注册、教师学生信息管理、在线考试、管理员登录及管理模块,以减轻教师的工作负担、提高工作效率和考试质量作为设计目标进行了较深入的研究与分析。

本文研究的重点在于考试系统的随机组卷策略,由于考核对象的不同,要求试卷内容也不一样,运用随机组卷策略才能解决这个问题。本文重点研究遗传组卷算法,分析了遗传算法的基本理论,对随机组卷问题进行了比较详细的阐述,形成了本考试系统随机组卷的数学模型,并在此基础上给出了算法的实现原理和代码描述。

3. 期刊论文 [胡滨](#)、[肖飞](#) [在线考试系统 -中国水运\(下半月\)](#)2008, 8(2)

本系统采用开放性的设计思想,题库可以随时进行增加、删除、修改等维护,包含题库管理和维护、智能组卷、试卷浏览、在线考试、阅卷评分、成绩查询等功能模块,并重点研究了其中的基于遗传算法的智能组卷方法。

4. 学位论文 [朱国际](#) [基于B/S在线考试系统的设计与实现](#) 2007

本系统采用基于MVC思想的B/S三层体系结构,设有系统管理员、考试管理员、领导、考生四种登陆方式,提高了系统的安全性。系统采用JSP-JavaBeans技术来实现B/S三层结构,其中用JSP来实现用户界面和表现逻辑,以JavaBeans来实现业务逻辑。组件技术的使用,大大提高系统的开发效率、灵活性和可移植性。同时还采用数据库连接池技术对系统的数据库访问连接进行了优化,通过使用遗传算法抽题组卷提高了系统组卷的成效。系统已经成功运行,极大地提高了柳州海关干部人事考核的工作效率,得到了用户的好评,其各项功能和性能都达到了预先的设计目标,具有易于维护、代码重用率高、非常灵活等特点。

本文介绍了本在线考试系统的开发背景,在对传统考试与在线考试方式与质量进行优劣对比的基础上,提出了利用现代化的计算机网络技术、数据库技术进行开发海关在线考试系统的必要性和可行性。并对整个在线考试的逻辑体系结构进行了流程分析和概要设计介绍,同时根据作者负责的考试系统数据库及B/S模式下考生子系统的设计与实现工作重点介绍了系统实现的一些关键技术。最后对系统中所存在的问题和需要进一步完善的工作进行了探讨和展望。

5. 学位论文 [贺东风](#) [基于CORBA三层C/S模式在线考试系统的设计与实现](#) 2009

当今信息时代,计算机得到了广泛普及和应用,计算机的使用已经渗透到社会的各个角落,成为人们生活中不可或缺的重要工具;人们对掌握计算机知识的需求与日俱增。计算机基础系列课程是实践性强的学科,操作能力在计算机基础课程中占重要的地位,为了便于上机考试,提高学生的操作能力,在计算机基础教学的各个层次中都应该引入机试。因此如何利用计算机网络,实现在线考试就显得尤为重要。随着计算机网络的普及、数据库技术的飞速发展以及基于网络的业务不断增多,传统的两层结构客户/服务器模式的应用系统暴露出在运行效率、网络安全和系统升级能力等方面的局限性。新的软件体系结构的推出成为一种必然,在这种情况下,多层分布式结构便应运而生。

本文深入分析了三层C/S模式的优缺点,并介绍了三层C/S模式的原理,以及多层分布式数据库应用系统的逻辑结构、实现技术、CORBA规范以及基于CORBA的多层分布式系统的实现方法。论文的目标是采用三层C/S模式实现一个适用于计算机基础教学的在线考试系统;因此论文详细论述了三层C/S模式在线考试系统的设计和实现过程。采用了面向数据流的开发方法,系统可分为三个子系统:考生信息子系统、考试子系统和管理子系统。系统对考试的全过程进行管理和监控;如考生报名、参加考试、试卷的提取、试卷的提交,试卷的批改,以及对题库的管理和维护;其中考试子系统是整个系统的核心部分,该系统采用三层C/S模式;考试系统的成功实现,组卷是关键的一部分,论文深入分析遗传算法技术,利用遗传算法实现考试系统的自动组卷;为确保考生帐号考试期间的唯一性,该系统的安全策略采用IP绑定技术、数据加密及负载均衡技术,从而提高了系统的效率、健壮性和安全性。

6. 学位论文 [钱成](#) [在线考试系统的设计与实现](#) 2003

该文论述的在线考试系统是为适应各种考试机构进行在线考试需要而设计的,可实现题库管理、自动智能组卷、在线考试、自动阅卷评卷等功能。文中论述了在线考试系统的体系结构、系统的主要功能和相关开发环境,详细阐述了该系统中对象提取定义以及VB环境下模块的实现方法。

7. 期刊论文 [赵玉兰](#)、[许薇](#)、[姜春风](#)、[ZHAO Yulan](#)、[XU Wei](#)、[JIANG Chunfeng](#) [基于C#在线考试系统的应用 -吉林农业科技学院学报](#)2009, 18(1)

在分析高校对在线考试系统及题库的需求、难点及解决方案的基础上,对系统的开发平台C#和.NET的适用性以及相关技术进行了分析,重点论述了系统的关键技术、数据库连接、组卷、信息安全等问题,实际有效的设计可为同类系统开发提供参考。

8. 学位论文 [吕大穷](#) [基于J2EE的智能在线考试系统的设计与实现](#) 2006

随着高校规模的扩大,招生人数的增加,如何改进教育教学手段,优化教学资源,提高教学质量是人们在不断探索的课题。而将网络信息技术、计算机技术、人工智能等先进技术运用到教育教学中来,是改革现代教育模式和手段的必然发展趋势。考试作为教育测量的有效手段是教学环节必不可少的一环。实现考试的网络化、智能化成为近年来人们研究的热点,在此背景下,本文作者运用先进的J2EE架构体系研究开发了一个能够进行多学科课程考试的智能在线考试系统。

该系统充分发挥网络优势,采用B/S结构,无须客户端,可以在机房、多媒体网络教室组织正式的考试;也可在联网(局域网或Internet)的任何一台计算机上进行考试。若设成练习、自测模式,学生则可以方便地利用考试资源进行学习和检测自己的学习情况。系统主要实现了题库管理、智能组卷、自动判卷和人工判卷相结合、网上查分,考试结果分析等功能。

本文从智能在线考试的实际需求出发,详细分析研究了基于测试理论的题库模型设计方法,对智能在线考试系统应具备的主要功能进行了详细论述。通过对智能在线考试系统组卷算法的重点研究,设计和实现了基于遗传算法的智能组卷。

9. 期刊论文 [潘晓辉, Pan Xiaohui 在线考试系统中组卷技术的研究 -价值工程2010, 29\(14\)](#)

组卷模块能否高效率地生成符合系统要求的试卷是评估考试系统的重要因素,遗传算法(GA算法)近年在解决多重约束问题的求解中应用非常广泛,把遗传算法应用于自动组卷,在标准遗传算法的基础上进行了改进,采用分段编码,基于误差的适应度函数和动态交叉及变异概率,采用遗传算法的智能组卷系统既能很好的满足组卷的约束,且效率远高于传统的回溯算法。

10. 期刊论文 [肖自红, 张祖平, Xiao Zihong, Zhang Zuping 基于.NET的试题库与在线考试系统研究 -计算机与数字工程2008, 36\(3\)](#)

在分析试题库及在线考试系统的需求、难点及解决方案的基础上,对系统的开发平台如.NET的适用性以及相关技术选型进行了分析,重点论述了系统的关键技术,提出了用二进制与图片文件相结合的存取方式以解决公式、图片等多种试题资源的入库与组卷等问题,论文还提出了采用遗传算法解决优化组卷、数字水印解决资源安全等关键技术,实际有效的设计与实现技术及具体的实例分析,可为同类系统开发提供参考。

本文链接: http://d.g.wanfangdata.com.cn/Thesis_Y1417375.aspx

授权使用: 李建平(wfnckdx), 授权号: 9bb9076f-c588-4f9f-b150-9de001410565

下载时间: 2010年8月28日