信息系统类毕业设计规范

（讨论稿）

1. **概述**

毕业设计（论文）是本科生毕业前学习、实践、探索和创新相结合的综合教学，是对学生综合运用所学知识解决本专业实际问题能力的考核。信息系统是常见的一类毕业设计，也是企事业单位应用最多的一类软件系统。

**信息系统**的主要目的是以一定的规范收集、整理企事业单位的业务数据，规范企业的业务流程，同时为特定的人员和业务提供数据的查询、统计、报表等功能的计算机软件系统。一个信息系统设计得成功与否，主要的标志在于是否满足了企业的业务需求，解决了企业在实际业务操作过程中的问题；而不是实现了的多少个增、删、查、改的功能。

随着软件开发技术的发展，信息系统中的数据收集、修改、查询等技术越来越简单。但企业的业务需求却越来越复杂，如多用户、细粒度的角色及权限控制、高并发、海量数据下环境下的查询与统计、业务工作流变化频繁等。因此，信息系统开发中的**关键问题**应在如何有效解决**特定环境下**的企业业务需求，该类毕业设计的论文中也应主要反映出企业**业务需求的特定环境、业务流程、解决方法、解决效果**等。

因此，这里的信息系统特指企业的信息管理系统。

1. **编写依据**

* 工程教育认证标准（通用标准及计算机类专业补充标准）
* 西南科技大学本科毕业设计（论文）撰写规范（2008年修订）
* 精简并行过程（Simplified Parallel Process）文档规范

1. **信息系统类毕业设计（论文）撰写规范**
   1. **标题**

标题应该简短、明确，能反映出系统设计的主要内容以及特点。标题字数要适当，一般不宜超过20个字。如果是某个大规模系统的子系统，为避免冗长，可以设副标题。

标题形式为“XXXX系统（平台）的设计与实现”、“XXXX系统（平台）开发（研发）”。

* 1. **论文摘要**

论文摘要以简练的语言概括设计系统的背景（要解决的问题）、设计的功能模块、采用的关键技术，以及对系统的测试方法及结论，应能反映论文中设计系统采用什么技术，解决了什么问题，并且达到了预期效果。摘要包括**中英文摘要**，字数一般应控制在300字左右，英文摘要应翻译准确、表述清晰。

* 1. **目录**

目录按三级标题编写（即：1……、1.1……、1.1.1……），要求标题层次清晰。目录中的标题应与正文中的标题一致，附录也应依次列入目录。

* 1. **正文**

正文是所设计系统中所有技术的详细表述，要让阅读论文的读者了解所设计系统的实现细节。其内容应当包括：**绪论（前言或引言）、系统需求分析、系统概要设计、系统详细设计与实现、系统测试与分析及结论**等部分。

* + 1. **绪论（前言或引言）**

绪论应说明本系统设计的需求背景、设计的意义、当前类似系统的现状、系统的设计目标及范围、要达到的技术要求及系统的实施方案等。

**需求背景**要说明企业为什么要开发此系统；**设计的意义**说明该系统能为企业带来的具体好处；**当前类似系统的现状**简述类似在国内外的发展概况及存在的问题，现有系统是否能满足企业需求，进一步说明需要定制开发该系统；**系统的设计目标**要说明系统最终要达到的效果；**技术要求**中要明确出系统要达到的技术指标，如数据库容量的大小（是否采用分布式）、系统在特定数据规模下的响应速度、最大支持的并发数、支持的客户端类型、兼容性要求、操作系统的版本、数据库系统的版本等，需要有具体的、可量化的参数。**系统的实施方案**简要给出如何完成该系统的设计，如何组织实施，一些额外资源如何获取等。

* + 1. **系统的需求分析（需求规格说明）**

需求分析部分应详细描述出系统的功能性需求及非功能性需求，明确系统的需求规格，让读者知道系统最终要实现的功能模块、业务流程及性能指标。需求规格说明应包含如下内容：

**系统介绍：**（1）说明系统是什么，什么用途。（2）介绍系统的开发背景。

**系统面向的用户群体：**（1）描述本系统面向的用户（客户、最终用户）的特征（如使用本系统的频率、计算机操作的熟练程度）；（2）说明本系统将给他们带来什么好处？他们使用本系统的可能性有多大？

**系统应遵循的标准或规范：**阐述本产品应当遵循什么**标准、规范**或**业务规则**（Business Rules），违反标准、规范或业务规则的产品通常不太可能被接受。如行业应遵循的**国家标准或规范**、**企业内部的业务操作规范**、**信息系统管理规范**、**信息系统密级管理规范、网络管理规范**等。

**系统中的术语与缩写解释：**系统中所涉及到的专业术语或缩写词，要给出详细的解释，便于不熟悉该行业的读者了解其中术语和缩写词的含义，不至于导致误解。

**系统的功能性需求：**描述出系统的整体功能需求及每个小功能的详细需求，先粗分，再细分。系统的整体功能需求可使用**整体用例图**来描述，让读者一看就了解到系统提供了那些功能；然后需要详细给出每个功能的详细需求说明，该说明中应详细描述该功能的**操作员、前置条件、后置条件、事件流（操作流程）、异常事件流、涉及的字段（数据）、操作业务的规则**等信息，可以用表格结合用例图、数据流图、UML活动图或时序图来描述。

**系统非功能性需求：**描述系统的用户界面、软硬件环境、系统质量及其他不属于功能性的需求。用户界面通常包含色调、浏览器版本、客户端类型等；软硬件环境通常指服务器、客户端最小支持的硬件参数、软件系统版本等信息；系统质量需求指正确性、健壮性、可靠性、性能（效率）指标、易用性、清晰性、安全性、可扩展性、兼容性、可移植性等，上述指标中应能进行量化。

**上述的功能性需求和非功能性需求给出的指标参数，在系统的测试部分应能得到有效的验证。**

* + 1. **系统概要设计**

系统概要设计主要描述出系统的顶层设计，主要包括体系结构设计，也可以包含界面设计和模块设计。

**体系结构设计中主要应包括：**

**（1）设计策略**：根据系统的需求与发展战略，确定设计策略（Design Strategy）。例如：**扩展策略。**说明为了方便本系统在将来扩展功能，现在有什么措施。**复用策略。**说明本系统在当前以及将来的复用策略。**折衷策略。**说明当两个目标难以同时优化时如何折衷，例如“时－空”效率折衷，复杂性与实用性折衷。

**（2）系统总体结构：**将系统分解为若干子系统，绘制物理图和逻辑图，说明各子系统的主要功能。说明“如何”以及“为什么”（how and why）如此分解系统。说明各子系统如何协调工作，从而实现原系统的功能。

**（3）每个子系统的结构与功能：**将子系统N分解为模块（Module），绘制逻辑图（如果物理图和逻辑图不一样的话，应当绘制物理图），说明各模块的主要功能。说明“如何”以及“为什么”（how and why）如此分解子系统N。说明各模块如何协调工作，从而实现子系统N的功能。

**界面设计**主要描述设计的规范、界面之间的关系图和工作流程图、主界面和每个子界面的设计、界面布局及色彩搭配的理由、界面设计的资源等。

**模块设计**主要描述系统中的各个模块的命名规则及每个模块的功能、接口、数据结构、算法等描述。

* + 1. **数据库设计**

数据库设计是信息系统设计中的核心部分，应详细描述系统中所有数据的存放规范。因此主要包含如下内容：数据库环境说明、命名规则、逻辑设计（E-R图）、物理设计（表设计）、安全性设计等。

**数据库环境说明：**说明所采用的数据库系统，设计工具，编程工具等详细配置。

**命名规则：**完整并且清楚的说明本数据库名称及表名、字段名的命名规则。如果本数据库的命名规则与机构的标准不完全一致的话，请作出解释。

**逻辑设计：**说明数据库中的实体关系图（ERD）。如果采用面向对象方法（OOAD），这里实体相当于类（class）。

**物理设计：**主要是设计表结构。一般地，实体对应于表，实体的属性对应于表的列，实体之间的关系成为表的约束。逻辑设计中的实体大部分可以转换成物理设计中的表，但是它们并不一定是一一对应的。对表结构**进行规范化处理（第三范式），要能有效分析出所设计的数据库表格所满足的范式**。因此，这里首先对表格汇总，然后给出每个表格的详细字段描述，**同时要进行范式分析**。

**数据库安全设计：**提高软件系统的安全性应当从“管理”和“设计”两方面着手。这里仅考虑数据库的安全性设计。如：

**（1）防止用户直接操作数据库的方法：**用户只能用帐号登陆到应用软件，通过应用软件访问数据库，而没有其它途径操作数据库。

**（2） 用户帐号密码的加密方法：**对用户帐号的密码（这里是指数据库系统的用户和密码）进行加密处理，确保在任何地方都不会出现密码的明文。

**（3）角色与权限：**确定每个角色对数据库表的操作权限，如创建、检索、更新、删除等。每个角色拥有刚好能够完成任务的权限，不多也不少。在应用时再为用户分配角色，则每个用户的权限等于他所兼角色的权限之和。

**优化：**分析并优化数据库的“时－空”效率，尽可能地“提高处理速度”并且“降低数据占用空间”。（1）分析“时－空”效率的瓶颈，找出优化对象（目标），并确定优先级。（2）当优化对象（目标）之间存在对抗时，给出折衷方案。（3）给出优化的具体措施，例如优化数据库环境参数，对表格进行反规范化处理等。

**数据库管理与维护说明**：在设计数据库的时候，及时给出管理与维护本数据库的方法，有助于将来撰写出正确完备的用户手册。

* + 1. **系统详细设计与实现**

系统的详细设计与实现主要给出系统几个核心关键问题的实现细节及效果即可，因此对系统中每个关键功能，应完整描述功能需求、功能的实现流程（程序流程图、时序图）、关键代码、实现效果等内容。

* + 1. **系统测试**

系统测试是验证系统实现是否满足前期设计指标需求，因此需要针对需求规格说明中的功能性需求和非功能性需求来进行测试。

功能性测试主要通过每个功能的测试用例来说明，因此应详细描述功能**测试的方法、测试用例、测试结果、测试结论**等内容，对测试的结果要进行**有效的分析**，说明是否达到了需求定义中的功能要求，达到的百分比，采用多少用例进行测试，用例的覆盖度是多少，未达到要求的原因是什么等。

非功能性测试包括用户**界面测试、压力测试、信息安全测试、可靠性测试**等，用户界面测试主要看是否满足用户的界面需求，通常为浏览器兼容性测试、色调、布局测试等；压力测试主要看系统能够是否满足了响应时间、高并发下的是否提供稳定的服务等，通常需要通过测试软件来进行；信息安全测试主要看系统是否满足了信息安全方面的需求，如SQL注入、数据库的安全性等，通常使用测试用例来说明；可靠性测试主要看系统能长时间运行过程中发生故障的概率，如可通过测试某个功能无故障运行的平均间隔时间、最小时间、最大时间来说明。除此之外，还可以有安装（部署）、反安装测试，即在不同软硬件环境下测试系统的安装（部署）与反安装的情况，通常给出实际的用例来说明。

对非功能性测试要能对测试的用例、数据进行有效的**汇总和分析**，验证测试结果是否满足了需求定义中的各种性能需求，并说明未达到的原因等。

* 1. **结论**

结论是毕业（设计）论文的总结，不是论文主体各章小结的叠加或简单重复。结论应准确、完整、明确、精练地阐述自己的工作或新的见解及其意义、作用，还可以进一步提出需要讨论的问题，提出建议或研究设想等。

通常描述论文详细描述的内容及作者的工作，如设计主要解决了什么问题，取得了哪些成果，那些问题是还没有解决的，对系统后期还有什么展望等。不要提与设计系统无关的内容，如学到了什么技术或自己的感想之类。

* 1. **致谢（可选）**

谢辞应以简短的文字对在系统设计或论文编写过程中曾直接给予帮助的人员（例如指导教师、答疑教师、同学、系统设计中提出建议的人员等）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。

* 1. **参考文献**

参考文献是论文不可缺少的组成部分，它反映论文的取材来源和广博程度。论文中要注重引用近期发表的与论文工作直接有关的**学术期刊类文献**。

产品说明书、各类标准、各种报纸上刊登的文章及未公开发表的研究报告（著名的内部报告如PB、AD报告及著名大公司的企业技术报告等除外）等不宜做为参考文献引用。但对于工程设计类论文，各种标准、规范和手册可作为参考文献。

引用网上参考文献时，应注明该文献的准确网页地址，网上参考文献不包含在规定的文献数量之内。

**各种程序设计类的教学书籍不宜作为参考文献**，参考文献格式参照“西南科技大学本科毕业设计（论文）撰写规范（2008年修订）”的相关规定。

* 1. **附录(可选)**

附录是指一些不宜放在正文中，但有参考价值的内容，例如公式的推演、编写的关键程序、图纸（用例模型图）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，若附录篇幅超过正文，会让人产生抓不住重点的感觉。

知识工程与教育技术团队

2016年1月

附：工程教育认证标准中毕业要求指标点与毕业设计符合度评价表（来自北京科技大学自评报告）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **“毕业设计(论文)”评分表设计** | | | | | | |
| 毕业要求指标点 | 指标点要求 | 1：非常不满意  （0.1） | 2：不满意  （0.3） | 3：一般满意  （0.5） | 4：满意  （0.7） | 5：非常满意  （0.9） |
| 2-7 文献研究与技术跟踪 | 针对复杂问题需求，能应用文献研究进行技术跟踪和现状综述 | 文献数量太少（<5）；或有少量文献（<10），无引用与现状综述 | 有部分文献并进行了引用（<10），且对相关理论和技术的发展、现状进行了简单说明 | 有足够的文献并进行了引用（≥10），且对相关理论和技术的发展、现状进行了基本的综述 | 有足够的文献并进行了引用（≥10），且对相关理论和技术的发展、现状进行了较充分的综述和总结 | 有足够的文献并进行了引用（≥10），且对相关理论和技术的发展、现状进行了较系统的综述和总结，并对其中的关键技术进行了分析 |
| 2-8 复杂计算问题定义、分析与建模 | 能分析建立复杂计算系统模型 | 涉及的计算问题太简单，或涉及复杂计算问题极少 | 涉及了少部分复杂计算问题，完成了初步的建模工作 | 对具有一定复杂性的问题，完成了建模和分析工作 | 针对复杂问题或应用系统，完成了建模和分析工作 | 对复杂问题或应用系统进行了较深入的分析，选择并建立了相应的计算模型 |
| 3-1 问题调研与需求分析 | 复杂计算系统确认和需求分析 | 问题调研太简单，需求分析工作极少 | 有一定的问题调研与确认，但需求分析简单 | 完成了复杂问题或应用系统基本的调研工作，并进行了一定的需求分析 | 对复杂问题或应用系统进行了较充分的问题调研和需求分析工作 | 针对复杂问题或应用系统，进行了较全面的问题调研和需求分析 |
| 3-7 复杂计算系统设计与实现 | 能从系统的角度权衡复杂计算问题所涉及的用户、技术、社会等因素，提出解决方案，体现创新意识，完成系统设计、实现，并通过测试或实验分析其有效性 | 没有从系统角度来考虑和分析问题，或涉及内容极少 | 初步完成了系统的设计与实现 | 从技术、用户、社会等方面进行了初步的综合考虑，完成了具有一定复杂性系统的设计与实现 | 从技术、创新、用户、社会等方面进行综合考虑，完成了复杂问题或应用系统的设计与实现，并对结果进行了验证 | 能从系统的角度全面权衡复杂计算问题所涉及的用户、技术、社会等因素，提出解决方案，体现创新精神，较好地完成了系统设计、实现，并通过测试或实验分析其有效性 |
| 4-3 实验综合分析 | 能对复杂工程问题相关模块实验或测试结果进行分析，并通过信息综合得出合理有效的结论 | 实验验证工作没有完成，或涉及的实验综合分析极少 | 进行了初步的系统与实验对比，完成了少部分实验分析 | 构建了计算系统，完成了基本的实验分析与综合对比 | 考虑了多种信息，进行方法、系统与实验对比，构建了相应的计算系统 | 基于较全面的信息综合，进行方法、系统与实验对比，构建了有效的计算系统 |
| 5-4 开发环境选择与使用 | 合理选择恰当的方法、技术、设备与工具，并应用于复杂计算系统研发 | 开发工具或环境涉及太少、太简单 | 使用了少部分简单开发工具与环境 | 使用了基本的开发环境 | 选择、使用了恰当的开发环境，并对其合理性进行了阐述 | 针对具体问题和实际应用，合理选择了恰当的方法、技术、设备与工具 |
| 8-3 计算机工程职业素养 | 能够在计算机工程实践中理解并遵守工程职业道德和规范，履行责任 | 没有涉及计算机工程职业素养，或者涉及内容极少 | 涉及了少量的计算机工程职业素养 | 在项目或应用系统中，体现了一定的计算机工程职业素养 | 在项目或应用系统中，针对具体问题分析时考虑了计算机工程职业素养相关内容 | 在项目或应用系统设计和实现过程中，较全面地考虑了计算机工程职业素养相关内容 |
| 9-3 信息共享与团队合作 | 项目相关的团队交流与合作 | 体现不了团队交流，或涉及极少 | 在某几个点上体现了团队交流 | 体现了一定的团队交流与合作 | 基于团队交流与合作，较好地完成了项目任务 | 处处体现了团队交流和合作的行为和效果 |
| 10-2 外语能力与国际视野 | 专业英语翻译与新技术跟踪 | 没有专业英语翻译，无新技术综述或涉及内容极少 | 少量的专业英语翻译，少量的新技术综述 | 完成了专业英语翻译，并对新技术进行了综述 | 良好的专业英语翻译、新技术跟踪与综述 | 较准确的专业英语翻译，并对新技术进行了较全面的跟踪、总结和分析 |
| 10-3 沟通与交流实践 | 撰写报告和设计文档、陈述发言、回答问题 | 报告格式不规范，内容太少 | 报告格式基本规范，内容不充实 | 报告格式基本规范，内容基本充实，回答问题和表述基本清楚 | 报告格式基本规范，内容充实，回答问题表述清楚 | 报告格式规范，内容充实，分析问题较透彻，回答问题表述清楚、正确 |
| 11-2 项目管理应用 | 在项目与工程中，考虑并适当应用工程管理原理与经济决策方法 | 没有考虑工程管理原理与经济决策方法，或涉及极少 | 少部分的地方涉及了工程管理原理与经济决策方法 | 对工程管理原理与经济决策方法进行了考虑和基本应用 | 在项目与工程中，对工程管理原理与经济决策方法进行了考虑和应用 | 在项目与工程中，对工程管理原理与经济决策方法进行了全面的考虑和适当的应用 |
| 12-2 终身学习与发展实践 | 能对解决的工程问题所采用的技术或开发的系统进行总结及展望分析 | 没有技术及系统的总结与展望分析，或涉及极少 | 涉及到了少部分技术或系统总结及展望，但不够恰当 | 对技术或系统进行了一定的总结和展望 | 对技术或系统进行了恰当的总结和展望 | 对技术或系统进行了较全面的总结和较深刻的展望 |