

芝士架构/系分论文宝典

适用 2025 年 11 月 - 芝士架构凯恩编辑整理 v3.0.0



目录

1. 前言	3
2. 论文科目总则	3
3. 论文批改说明	4
4. 论文考试基本常识	5
4.1. 难度统计	5
4.2. 答题方式	6
4.3. 考试时间	6
4.4. 评分标准（官方）	7
5. 论文字数分配说明	8
6. AI 论文助手辅助	9
7. 项目背景与 PRD 文档	10
8. 摘要写作技巧（300 字）	13
9. 正文写作技巧（2500 字）	14
9.1. 项目背景（500 字）	14
9.2. 技术方法说明（500 字）	15
9.3. 论点展开（1000 字）	15
9.3.1. 句式方面	16
9.3.2. 两段论	17
10. 结尾写作（400 字）	18
11. 论文范例（必读必看）	18
12. 论文自查清单	23

版权提醒:本资料已通过国家版权局登记(登记号:渝作登字-2025-A-00624260),
(一旦发现资料恶意流出,直接封号不退款,后果严重的话会请求平台披露你的个人信息,并于互联网法院起诉,请珍惜账号权益和个人名誉,谢谢)。假如发现有机机构或者跟个人盗版可以第一时间联系我,我会及时跟进投诉处理,并承诺给红包返现。

1. 前言

此论文宝典的初衷是帮你梳理论文写作中的写作技巧,备考方法。为了避免有的同学直接背这里的范文不去动脑筋。我这里以软考阅卷系统为例,斗胆设计这个主题的论文,说明如何去结合一个产品/项目来构思一篇软考作文。这个资料主要聚焦备考的流程和组织材料的方法,而不是给出具体的可以让你直接背的范文,想要论文押题和核心论点的,等我红宝书论文押题公布。

2. 论文科目总则

软考论文如何拿高分?秘诀就在于差异化。论文考试其实是主观的,判卷人一天要改上百篇文章,怎么让他觉得你的内容是 OK 的?是你的项目有多高大上?你用的技术有多高端?是看你供职于大厂还是小企业?错得离谱,大错特错!

实际上阅卷人是看在他看过的这些内容里,你的文章是否在构思上有差异性,差异性包括是否有深度,是否有自己的思考,你对项目的理解是否到位,你站在什么位置来阐述你的内容,以及是否过于模版化 AI 化、流水线化(后面会讨论)等等。

凯恩这里所有的写作策略也都是为了让你奔着差异化去的,目的就是让你能

够获得比较优势。一旦此法又被曝光，那么所谓的差异化就会更困难，考试实战效果也就越差。试想假如大家都按照我这里的套路来写，那么等于大家又站在同一起跑线上，你想要出挑就更加困难。出于这个考虑，我想告诉所有看到这份资料的同学履行下面约定：

第一，我不希望你把我这里内容随意分享出去。

第二，不要背这里的范文。尽管这里的范文水准很高，但因为已经有曝光度，你抄我也抄，最后大家写的都一样，又违背了差异化原则。所以千万不要把它作为考场上论文的最终呈现。一定要形成自己的叙事风格，接地气，去模板。最好的状态是，考场凭自己的理解来写，这样才能独树一帜，才能过。

第三，论文宝典里的范文行文方式一定要好好体会，把手弄脏，亲自去写去练。论文听人（包括凯恩在内）吹牛怎么写是没用的，第一步就是仿写，仿思路，仿格式来写，这里的范文就是最好的素材。第一篇很难，可能要花你 2 天时间，这里的时间花费主要在于背景的整理，项目的复盘，项目的选择，以及可能的部分知识的查阅和学习，不要慌，此时想得越多写得越多你收获就会特别大！

3. 论文批改说明

每次考试前，我都会给出本年度的论文押题（不是本份资料，架构和系分是单独分开的）。你提交给我批改的论文一定要参照这些题目进行练习，其他题目原则上不进行批改。这是因为，论文考试中，既有的题目再考的概率很低。要考也是往往会做一些变化。假如你没有研究，看到什么练什么，对你复习备考没有好处，白白浪费时间而已。

另外由于都是凯恩人工批改，业务高峰期（考前一个月），响应可能没有那

么快，原则上需要 2-3 天的时间，希望耐心等待。各位可以直接以 Word 形式发送文章到我的微信上 deckardcain2 这个微信号。我会以 Word 修订方式对你的文章进行指导，然后发还给你，并且告诉你该文章得分。一定要用电脑打开我发给你的 Word 才能看到批注，有的手机不支持直接看批注。

一般来说，假如我给你的分数只有 45 分以下，问题肯定是很大的，一定要自己再去好好改改。改好再发给我进行确认，直到通过为止！这样对你的提高是最大的。2025 年 11 月考期论文提交截止日期 11 月 1 日晚 24 点！

4. 论文考试基本常识

4.1. 难度统计

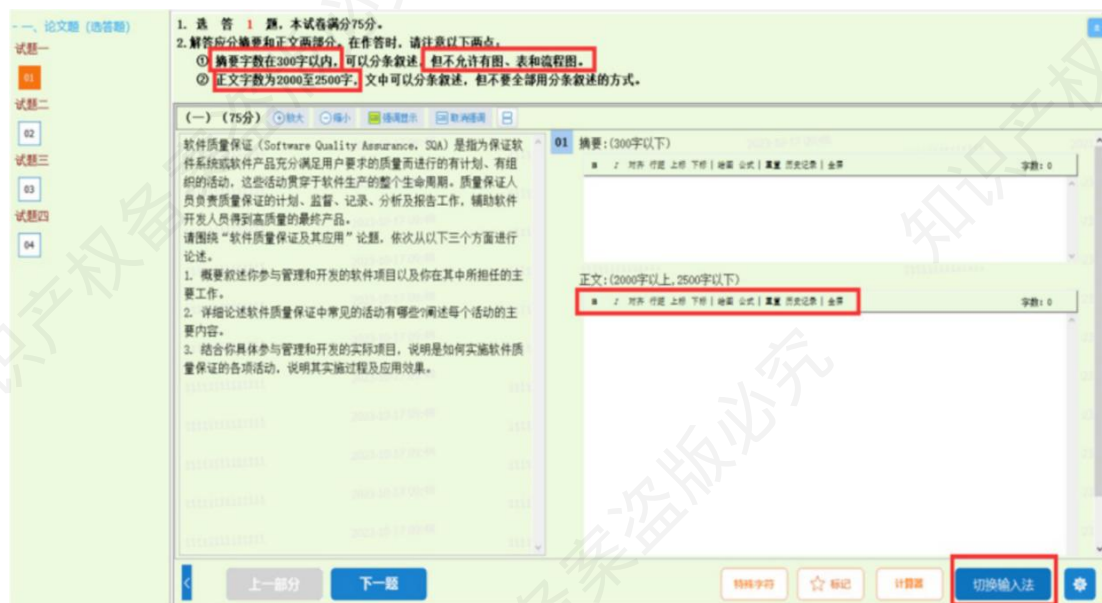
论文写作通过率数据如下表所示，这是针对芝士架构会员内部的统计数据。结果表明，论文通过率基本在六至七成左右。这个数据比高项系规的通过率高很多（三至四成）。

考试科目	系统架构设计师	系统分析师
2022 年 11 月	80%	没有举行考试
2023 年 05 月	没有举行考试	76%
2023 年 11 月	70.97%	没有举行考试
2024 年 05 月	70.5%	69.5%
2024 年 11 月	64.2%	83.8%
2025 年 5 月	62.74%	70%

但是要说明，论文阅卷主观性很大，非常大，取决于你和其他人的内容给阅卷的感官，可能你觉得自己写的还不错，但是和别人一比要么是内容不行，要么过于套路，要么是字数差点意思。论文高低是比出来的，这个一定要清楚。

4.2. 答题方式

论文科目也是上机考试。和以往笔试相比，上机考试的优势很明显，最主要的是同学不用担心字迹潦草，字迹歪歪扭扭导致卷面失分了。你只要专注构思内容就行。机考论文界面如下图所示。论文考试机考，常用的输入法基本支持。



这里需要注意的是，论文分为摘要和正文两大部分（架构系分仍然是要写摘要的）。其中摘要部分需要严格控制在 300 字以内（包含标点）。这是因为这里的控件会检测字符数，超出 300 的字符是无法输入的。而正文部分一般要求 2000 到 2500 字之内，但超过 2500 仍然是可以输入的。这里，凯恩建议要遵照考试要求，严格控制正文字数，不要超过 2500。字数控制在平时练习的时候一定要多加注意。

4.3. 考试时间

论文写作时长为 2 小时，2024 年 5 月开始，论文放在下午单独考试。

4.4.评分标准（官方）

论文满分是 75 分，论文评分可分为优良、及格与不及格三个档次。评分的分数段可分为：60 分至 75 分为优良；45 分至 59 分为及格；0 分至 44 分为不及格。具体评分时，参照每一试题相应的“解答要点”中提出的要求，对照五个方面进行评分：

切合题意（分数占比一般为 30%）：可分为切合题意、符合题意、基本符合题意、偏离题意等档次。

应用深度与水平（分数占比一般为 20%）：分为很强的、较强的、一般的、较差的独立工作能力 4 档。

实践性（分数占比一般为 20%）：包括有大量实践和深入的专业级水平与体会、有良好的实践与切身体会和经历、有一般的实践与基本合适的体会、有初步实践与比较肤浅的体会 4 档。

表达能力（分数占比一般为 15%）：从逻辑清晰、表达严谨、文字流畅和条理分明等方面区分为 3 档。

综合能力与分析能力（分数占比一般为 15%）：分为很强、比较强和一般 3 档。

有没写论文摘要或摘要过于简略，过于自我吹嘘、夸大其词，有明显错误和漏洞等情况的可能会适当扣 5-10 分。有虚构情节、内容不可信或不真实，主要从书本知识和资料摘录进行讨论，内容不切题意或内容空洞，篇幅过短，条理不清晰等情况之一可能会被判定为不及格。

有以下情况之一的则会考虑适当加 5-10 分：有独特的见解；起点很高、符合新趋势并能初步实现；内容详实，思路清晰，非常切合实际；项目难度高或完

成的质量优异并能按要求论述等。

这里最重要的实践性真实性，这个是决定你是不是能及格。对于架构还有系分来说，除非你写的牛头不对马嘴，一般不会因为你某个名词解释的不准确让你不及格。

5. 论文字数分配说明

软考里对论文写作实际上有一定的要求，一般来说摘要不能超过 300 字，正文不能超过 2500 字。所以根据这个要求，我们一般根据考试要求，把整个论文结构做一个分解，具体如下表所示。

论文框架	内容	字数
摘要	项目简介，个人的工作，以及本文所要论述的内容。这里不要太模板化，要体现自己的一些技术认识和观点，参照下面的范例。	300
项目背景	项目简介，个人的工作，不过比摘要要更加具体，公司启动这个项目的战略依据，环境，投入的资源，工期等要做全面的论述。	500
技术方法说明	回应问题 2 （长度最好是正文的一半，这样详略处理比较好，突出主要内容）	500
论点展开	回应问题 3 根据论点分段 2 段为佳，接地气，用自己的话，自己的理解来写，你是 CTO，你是架构师。	1000
论文结尾	总结技术运用成果，项目效益，提出相关技术实践落地感悟，为进一步优化拓展打好基础。别写遇到的问题了，已经被写烂了，换个方式，写写你对主题的理解。	400

你必须根据上表的要求，合理安排内容的详略。

6. AI 论文助手辅助

以往论文准备很难，需要找资料、组织语言、结合项目，工作量很大，会掉很多头发。现在有了 AI 大模型，实际上这个工作就会很好做。凯恩设计的 AI 论文助手针对软考论文要求做了精细化调整，可以一次性生成完整的符合软考格式要求的文章。看到这篇文档的同学目前都有 15 次论文生成机会，点击进入 https://www.cheko.cc/little_cain，可以看到论文助手。



进入之后左边的方框输入你的论文主题，右边方框输入你的项目名称，点击

生成，AI 会在 3 分钟之内给你生成高质量的范文。

但是目前生成的范文字数可能超过考试需要。比如摘要可能不能做到正好 300 字。主体部分可能也会超过 3000 字。这个时候需要你自己去调整，让它符合我们前面提到的字数分配的要求。

← 云原生技术应用视角下的一体化政务服务平台系统构件设计与实现

复制 下载

注意事项：AI生成的内容可能出现字多或者例子不贴的情况，需要自己调整略做修改，请注意甄别细节

当前文章（包含摘要有） 3991 字。凯恩的要求是正文 2500 左右，摘要 300，不要超过。

摘要

在数字化政务改革的浪潮下，各地政府积极推动政务服务数字化转型，以提升政务服务效率和民众办事体验。一体化政务服务平台系统作为承载各类政务服务的核心枢纽，需处理海量的业务数据和复杂的业务流程。传统的信息技术架构在扩展性、灵活性和资源利用率等方面逐渐难以满足平台快速迭代和高并发访问的需求。为解决这些问题，我所在的项目团队决定引入云原生技术对一体化政务服务平台系统进行升级改造。

我在该项目中担任技术骨干，负责云原生架构的整体设计与关键技术选型。主导了容器化技术在政务服务应用中的落地，实现了应用的快速部署和弹性伸缩；推动了Kubernetes集群的搭建与管理，保障了系统的高可用性和负载均衡；参与了微服务架构的拆分与重构，提升了系统的可维护性和开发效率。通过云原生技术的应用，一体化政务服务平台系统的性能和稳定性得到显著提升，能够更好地应对业务高峰，为政务服务的数字化转型提供了坚实的技术支撑。

正立

7. 项目背景与 PRD 文档

通过近几年的辅导凯恩得出一个结论，就是决定论文好坏的不是技术，而是全局的系统化的产品战略思维。产品往往牵引着解决方案的。一个好的解决方案也往往和产品有这高度的匹配关系。所以第一步，是梳理你的产品。这里有个很绝的办法，就是让大模型帮你生成一份完整的 PRD 文档。

可能有的同学不知道什么是 PRD 文档，实际上 PRD 文档是 Product Requirements Document(产品需求文档)的缩写，是产品开发流程中核心的“需求交付物”，本质是将“产品目标”转化为“可执行、可验证的具体需求”，为研发、设计、测试、运营等跨团队角色提供统一的工作依据和判断标准。

既然是基准那么基准的好坏决定产品的高度,产品的高度决定了解决方案的高度,从而决定了你论文的好坏。这个逻辑要理清楚。如何获取 PRD 文档?你手头有当然最好,假如没有的,那么直接用 AI 生成。提示词如下:

你的任务是生成一份 PRD (产品需求文档)。请仔细阅读以下信息,并按照指示撰写文档。

产品名称:

<product_name>

软考 AI 评分系统

</product_name>

产品特性:自动阅卷,评分,考试难度分析,题目难度分析,可以和答卷扫描设备相连,人工介入等等

目标用户:

<target_users>

阅卷中心

</target_users>

在撰写 PRD 文档时,请遵循以下指南:

1. 文档应包含以下部分:产品概述、目标用户、产品特性、功能需求、非功能需求、验收标准。
2. 产品概述部分需简要介绍产品的名称和用途。
3. 在目标用户部分,要详细描述产品的目标用户群体特征。
4. 产品特性部分应清晰列举产品的各项特性。
5. 功能需求部分要详细描述每个特性对应的具体功能要求。

6. 非功能需求部分需涵盖性能、安全、易用性等方面的要求。
7. 验收标准部分要明确产品达到何种条件可认为通过验收。
8. 使用清晰、准确、专业的语言，避免模糊和歧义。
9. 确保文档内容完整、逻辑连贯。

请在<prd_document>标签内写下你的 PRD 文档。

此时你会拿到如下所示的产品需求文档。此时你会发现这里所有的功能都写的很完善，模块构建也很系统，基于这个产品需求基座，你的项目背景和解决方案肯定会比別人写的更加深入和贴切，这个文档保存好，后面精修论文的时候有大用。考场上也尽量围绕这个系统去编，一定程度上可以解决内容太假的问题。

软考AI 评分系统 — 产品需求文档 (PRD)

一、产品概述

软考AI 评分系统是一套面向阅卷中心的端到端智能阅卷与测评平台。系统通过与高速答卷扫描设备对接，完成答卷数字化、版面解析、客观题自动判分、主观题AI评分与人机协同复核；并基于测量学与统计模型输出题目难度、区分度、信度等考试质量分析，提升阅卷效率与一致性，降低人为误差与组织成本，满足大规模集中阅卷的安全合规与可追溯要求。

二、目标用户

- **省/市/校级阅卷中心**：具备集中扫描、封闭内网、严格保密要求与批量阅卷组织能力的机构。
- **角色画像**
 - **阅卷管理员 (Admin)**：创建考试、配置卷模板、分配权限、监控进度与质量、导出成绩。
 - **学科/题组负责人 (Leader)**：维护评分标准与示例，执行标定与一致性控制，审批争议与复评。
 - **阅卷员 (Grader)**：在人工介入界面执行复核/修订评分；处理模型低置信度与异常样本。
 - **质检员 (QC)**：抽检、双评一致性核对、偏差审计与报表出具。
 - **系统运维 (IT/Ops)**：设备接入、运行监控、日志审计、备份恢复与安全加固。
- **业务规模假设**
 - 单场次 1-10 万份答卷，峰值 60 页/分钟/扫描通道；题型包含单选/多选/填空/简答/论述/设计题。

三、产品特性

1. **自动阅卷与评分**：OMR 客观题自动判分；OCR/版面解析；主观题AI评分（含置信度与解释）。
2. **人工介入**：低置信度/争议/抽检样本的人机协同复核、双评与仲裁、复评申诉闭环。
3. **考试难度分析**：全场难度指数、信度 ($KR-20/\alpha$)、成绩分布、等值与异常检测。
4. **题目难度分析**：题目通过率 (p 值)、区分度 (点二列相关/序列相关)、选项诱异性、漏判/错判告警。

- FR-07 题块切分：版面自适应与歪斜矫正；题块、作答区、批注区结构化JSON。
- FR-08 元数据生成：每份答卷生成结构化记录（考生ID、科目、题目作答、影像指针、时间戳）。

C. 自动评分（客观/主观）

- FR-09 客观题判分：按标准答案与赋分规则自动出分；支持多选、组合题、填空多槽位匹配/模糊匹配。
- FR-10 主观题AI评分：基于评分细则与锚题进行标定；输出分数、置信度、关键证据/要点匹配解释。
- FR-11 置信度与路由：低于阈值（可配）自动路由由人工复核；相邻阈值进入双评池。
- FR-12 连坐/异常检测：同题作答高相似度、异常模式与重复答案检测，标记待审。

D. 人工介入与质检

- FR-13 复核工作台：按题/按人两种视图；快捷键、批注、模板短评；一键对比AI要点匹配。
- FR-14 双评与仲裁：支持双评、盲评；分差>阈值自动触发仲裁，提供一致性提醒与Kappa 指标。
- FR-15 复评申诉：按规则受理；记录修改前后差异与原因；生成复评报告。
- FR-16 质量看板：阅卷进度、平均用时、一致性、阅卷员偏差、抽检通过率与红黄线告警。

E. 考试/题目难度分析

- FR-17 考试级指标：难度（平均得分率）、信度（KR-20/ α ）、标准差、分数段分布、等值参考（可选IRT）。
- FR-18 题项级指标：p 值、区分度（点二列/序列相关）、干扰项选择率、信息函数（2PL可选）。
- FR-19 报表输出：一键生成《考试质量分析报告》《题目分析报告》；支持自定义模板与导出。

F. 评分标准与模型治理

- FR-20 评分细则库：按科目/年份/题型管理；示例作答（正/负例）与锚题；版本对照。
- FR-21 标定流程：抽样金标集创建（ ≥ 1000 题实例，科目可配）；拟合-验证-上线；回滚与A/B。
- FR-22 持续校准：监控人机差异、漂移与题型域偏；阈值自动调优建议与人工确认。

8. 摘要写作技巧（300 字）

摘要除了牢记 300 字的字数限制，内容上也要精心设计。

开篇点明项目的核心领域与大致目标，在这个部分，在内容上一定要突出你的行业特色。比如你是做电商的，你是做在线教育的，那么你所做的项目具体服务什么业务？为什么要做？在这里一定要交代清楚。例如 “在当今数字化转型浪潮下，[项目所属行业] 领域对 [项目核心技术或业务目标] 的需求日益增长，[项目名称] 应运而生”。

然后阐述自己在项目中的关键角色与承担的主要任务，像“笔者作为项目的关键推动者，负责从技术架构设计到团队协调等多方面工作”。

对于要论述的内容，提出独特的技术视角或创新点，如 “笔者经过什么样的思考，发现 [新技术或方法] 非常适合解决当下业务中遇到的什么瓶颈，困难

等。本文将深入探讨在项目中如何创新性地运用 [新技术或方法] 解决传统 [相关技术难题或业务痛点]，并分析其对项目整体架构与性能提升的显著影响”。

具体的成品请看第三部分论文范例摘要部分。

9. 正文写作技巧 (2500 字)

9.1.项目背景 (500 字)

项目背景这块看似简单，实则考察你的站位。这里的文字可以体现你对项目的整体理解是否到位。这里一种常见的，烂大街的套路是，这里我写我用了什么具体的框架，前端后端，恨不得写上十几种，而对项目的建设的目的意义背景避而不谈，典型的捡芝麻丢西瓜。所以凯恩建议，在开头部分一定要体现你作为公司中高层，如何看待这个项目的建设目的，这背后可以反映该项目对于公司来说有何重大意义，公司投入多少资源来做这个事情。而这些，就是你和那些写 SpringBoot, Vue, Redis 的考生在位面上的差距。另外在这个地方，你还要体现你项目包含了哪些基本的要素，模块，功能。这些内容尽可能是你业务场景所独有的。一般不建议写什么权限管理，用户管理这种通用性较强的模块，More More Low Low，写得越多分越低。

还记得前面的 PRD 文档吗？在 PRD 文档里我们可以看到产品概述，如下所示。实际上产品概况这里已经做了非常综合的说明，放在项目背景里会有很大提升。

软考 AI 评分系统是一套面向阅卷中心的端到端智能阅卷与测评平台。系统通过与高速答卷扫描设备对接，完成答卷数字化、版面解析、客观题自动判分、

主观题 AI 评分与人机协同复核；并基于测量学与统计模型输出题目难度、区分度、信度等考试质量分析，提升阅卷效率与一致性，降低人为误差与组织成本，满足大规模集中阅卷的安全合规与可追溯要求。

具体的成品请看第三部分论文范例项目背景部分。

9.2.技术方法说明 (500 字)

技术方法说明，凯恩有的时候又叫它回应问题二。也就是论文题目中的第二个字题目。以系统分析师 2023 年第 2 题论敏捷开发方法为例，它的第二个子题目是让你简要概述 Scrum 开发方法中的角色、工件和活动。

这个地方一般来说是有难度的。难度在于一般让你作答的内容是理论性的，概念性的，系统性的。99%的考生不是研究软件理论的，都是写代码搞定需求的，那么这个地方回答的往往不够完整。所以每次考试考生都会和我说，【凯恩，我问题二答得不好，感觉很多点没写到，是不是要挂了】。

好消息是，No，不会。因为 99%的人基本写不对写不全。除非这道题已经曝光很久被押题压中了，否则大家都一样，都会漏写。漏写没关系，这个章节唯一要注意的是，字数。字数决定态度，决定你是 45 还是不及格。建议水到 400 字，用实际行动告诉阅卷人，我尽力了，这次你让我过吧！

漏写不等于乱写。假如问题二你都不知道它在问什么，那么我建议你换一题。

具体的成品请看第三部分论文范例技术方法说明部分。

9.3.论点展开 (1000 字)

论点展开，凯恩有的时候又叫它回应问题三，这是整篇文章的核心。在这个区域，阅卷人就是会拿放大镜来看你，一旦你有一些泛泛而谈，空洞的语句，他

就要不由自主地控制他的小手给你不及格了。另外注意的是，不要輕易的去背范文，去背模板。这种范文模板曝光度太高了，内容就很容易在阅卷的时候被判定为雷同，被扣上不真实的帽子。

所以这里我们的目标就是精心构造我们的表述，我们的文字，让他们觉得我们理解得太深了，太接地气了，有比别人高出一筹的企业级理解。

什么叫做企业级理解。是指技术选型的思考，架构设计的初衷，解决什么问题。另外就是这个落地过程中你有什么新的理解等等。

另外在形式上，这里主要有两个基本原则需要注意，第一个是句式方面，另一个是要分两段论来写。

9.3.1.句式方面

那我们到底应该怎么做呢？毕竟想要达到企业级理解你得真懂才行，一般同学没有那么多体会，是不是肯定过不了了？不是的。这里第一步，就是通过添加关键的词语引导阅卷人，给他心理暗示，证明你思考过。这里凯恩总结了以下 4 种句式，请你在后面的练习中熟练运用。

句式 1	由于【业务场景的某种特殊性】，笔者深切感受到，【某种具体的新技术或者新方法】的设计是关键环境。在设计之初，我们错把【说一种错误的做法】，【导致了什么的结果】。通过深入分析，我们发现实际上，在我们的业务流程中【说一种正确的做法】更好，【它可以解决了什么问题】
句式 2	以【某种具体的新技术或者新方法】为例，我所在团队【说一种正确的做法】。并且出于什么目的，笔者还【说一种正确的做法】。
句式 3	在面对【业务场景的某种特殊性】时，我意识到【某种具体的新技术或者新方法】的重要性。在初步尝试中，我们错误地选择了【一种不恰当的策略】，【导致了效果不理想】。通过反思与调整，我们最终发现，采用【一种更为合理的做法】

	能更有效地【解决原本存在的问题】。
句式 4	在实际工作中，我逐渐理解到【某种具体的新技术或者新方法】的价值。通过实施【一种有效的策略】后，团队不仅提升了工作效率，还【解决了之前遇到的具体难题】。

9.3.2.两段论

这个部分之前套路是三段论，但是实践下来发现，三段论的最大问题就是 $1000/3=300$ 字往往说不透说不深，所以从今年开始，凯恩建议两段论会更好。两段落划分之后，通过两个角度互相印证，同时这样每个论点保持在 500 字，内容就可以写的很饱满，又可以把论点说透。

另外，每个论点要结合你的实际案例来写，这个大家现在都可以做到，比如举例子，“在笔者的项目中，我们团队用 xxx 缓存实现了热点功能暂存，提升读取效率，相比直接数据库读写，响应时间从 ms 降低到 ms，吞吐率从 xxx QPS 下降到 xxx QPS。”但是光这样写不够。因为太简单了，阅卷的看的太多了，没亮点没细节，容易判成低分。

什么是细节？你系统的某个功能，你调优的某个场景，你解决问题的思考过程，你的系统化的思维。进一步思考，你可以加哪些东西进去。这里凯恩建议从场景、数据量、运维角度切入，比如热点数据有哪些？在什么场景下读写效率要求无法达到要求了？热点数据规模有多大测算过吗？需要多大的虚拟机资源运维支持配置。另外可以从技术对比切入，你为什么要这么做？方案 B 为什么不行，从你业务的具体场景讲将你技术选型的思路等等。

具体的成品请看第三部分论文范例论点展开部分还有等我的论文押题宝典发布。

10. 结尾写作 (400 字)

结尾写作是套路模版的重灾区，想要贯彻凯恩的差异化原则，就是要突破传统模板，构造更高级的没有痕迹的模版。以往的模板是要写项目顺利上线得到一致好评，然后还有一点问题，后续我们会改进云云。这个烂大街的东西千万别用，至少芝士架构的学员不要这么写。

这里凯恩给出的思路是，结尾部分一开始讲讲根据你的工作/设计，达到了什么样的成效。然后复盘思考，在这个工作中，你对主题【新方法/新技术】有哪些更进一步的理解，这里要提出“暴论”，比如人人都说微服务好，你说不，微服务仍然有很多问题，运维监控复杂度上升，管理成本提高。分布式系统在问题排查等方面也引入了新的复杂度。尽可能从实践角度出发来理性看待新技术/新方法的问题，把这些问题抛出来，体现你的思考，这是在这一段要做的事情。而不要套路化的不管什么题目都用一个结尾，会让人反感。

具体的成品请看第三部分结尾写作部分。

11. 论文范例 (必读必看)

这是系统分析师 2023 年第 2 题论敏捷开发方法，这是凯恩按照上面的作答要求反复修改润色的内容，希望大家好好研读，做做笔记，为后续仿写打下基础。

敏捷软件开发遵循一套软件开发的价值和原则，在开发中，需求和解决方案通过自组织跨功能的团队达成。敏捷软件开发主张适度计划，迭代开发，提前交付与持续改进，并且快速灵活地应对变更，作为敏捷开发方法之一，Scrum 方法被广泛应用。请围绕敏捷开发方法论题从以下 3 点进行论述。

1. 概要叙述你参与管理和开发的软件项目，以及你在其中担任的主要工作。

- 2.请简要概述 Scrum 开发方法中的角色、工件和活动。
- 3.具体阐述你参与管理和开发的项目如何基于 Scrum 敏捷开发方法进行系统开发的。

针对这三个问题,我们要在论文里一一给出回应。具体字数安排见下方表格。

可以看到这里至少有 1200 字是确定的。这 1200 字包括 300 字摘要,500 字项目简介,400 字结尾。这些内容只要你好好准备,形成你自己的模板,结合论文主题,考场上小改改就能写出来。

在论文准备的时候,记得生成 PRD,并用 AI 论文助手生成长文作为参考。

在这个基础上改可以帮你节约时间。

生成记录

PS: 执行错误

← 敏捷开发视角下的软考阅卷系统构件设计与实现

复制 下载

注意事项: AI生成的内容可能出现字多或者例子不够的情况,需要自己调整略微修改,请注意甄别细节

当前文章 (包含摘要有) 3467 字。凯恩的要求是正文 2500 左右,摘要 300, 不要超过。

摘要

在软考行业,随着考试规模的不断扩大,传统的阅卷方式效率低下、易出错,已难以满足日益增长的阅卷需求。为了提高软考阅卷的效率、准确性和公正性,我司启动了软考阅卷系统的项目建设。我作为该项目的敏捷开发团队负责人,负责制定敏捷开发计划和流程,组织每日站会、迭代规划会议和回顾会议等。带领团队采用敏捷开发中的 Scrum 框架,快速响应需求变化,将系统开发划分为多个短周期的迭代。在每个迭代中,明确用户故事和任务,严格把控进度和质量。同时,积极与业务方沟通,确保系统功能贴合实际需求。通过敏捷开发模式,我们高效地完成了软考阅卷系统的开发,系统上线后显著提升了软考阅卷的整体效率和质量,为软考工作的顺利开展提供了有力支持。

正文

随着软考报考人数逐年激增,传统人工阅卷方式效率低、易出错,已难以满足大规模考试的评卷需求。公司管理层经过深入调研与战略规划,决定启动软考阅卷系统的开发项目,旨在借助先进的信息技术手段,提升阅卷效率和准

电商平台构件设计与实现

创建于 2025-08-14 13:07:39

算系统构件设计与实现

创建于 2025-08-13 15:21:46

创建于 2025-08-13 12:00:17

创建于 2025-08-13 11:58:36

摘要 300 字	随着考试规模的不断扩大,传统的阅卷方式效率低下、易出错的方式逐渐渐显,已难以满足日益增长的阅卷需求。为了提高软考阅卷的效率,我司启动了软考阅卷系统的项目建设。我作为该项目的敏捷开发团队负责人,负责制定
-------------	--

	<p>敏捷开发计划和流程，组织每日站会、迭代规划会议和回顾会议等。带领团队采用敏捷开发中的 Scrum 框架，快速响应需求变化，将系统开发划分为多个短周期的迭代。在每个迭代中，明确用户故事和任务，严格把控进度和质量。同时，积极与业务方沟通，确保系统功能贴合实际需求。通过敏捷开发模式，我们高效地完成了软考阅卷系统的开发，系统上线后显著提升了软考阅卷的整体效率和质量，为软考工作的顺利开展提供了有力支持。</p>
项目背景 500 字	<p>随着信息技术行业快速发展，软考作为核心职业资格认证考试，报考人数连续五年同比增长超 15%，2024 年报考规模更是突破 80 万人次。传统人工阅卷模式的弊端愈发凸显：单批次试卷需 20 余名专业阅卷人员耗时 2 周才能完成评阅，不仅效率低下，还因阅卷人员主观判断差异，导致评分误差率超 3%，既无法满足大规模考试的时效性要求，也对考试公平性构成挑战。</p> <p>为破解这一困境，公司管理层高度重视，牵头开展为期 3 个月的深度调研——不仅分析国内 10 余个教育测评信息化标杆案例，还与软考办进行 6 轮需求对接，重点明确阅卷效率、评分准确性、数据安全性等核心诉求，最终结合公司技术优势，确定启动软考阅卷系统开发项目。该系统围绕 “高效、精准、可控” 设计以下核心模块：试卷扫描录入模块支持高清高速扫描，单小时可处理 5000 份试卷，还能自动识别科目类型、校正页码偏差，避免人工整理失误；评分标准设定模块采用可视化配置界面，可灵活调整单选、多选、判断等不同题型的得分规则，且支持实时更新，适配软考大纲调整需求；智能阅卷模块依托自研 AI 算法，对客观题实现 99.95% 准确率的自动批改，同时标记疑似异常答案。项目整体建设周期 6 个月，是公司在教育信息化领域的重要实践，为后续同类项目积累了宝贵经验。</p>
技术方法说明 500 字	<p>笔者在软考阅卷系统项目中采用敏捷开发方法，主要是因为该项目需求可能随考试要求和政策变化而变动，且需快速响应并交付可用系统。敏捷开发强调团队协作、客户参与和快速迭代，能有效应对这些挑战。敏捷开发以用户故事为核心，将系统需求拆分为小而可管理的任务，便于团队理解和执行。在本项目中，我们通过用户故事明确了如试卷录入、评分规则设定、成绩统计等功能需求。其采用短周期迭代方式，每个迭代都有可交付成果，能及时获取用户反馈并调整方向。例如在每个迭代结束后，邀请阅卷老师试用系统，根据他们</p>

	<p>的意见优化功能。同时，敏捷开发注重面对面沟通，每日站会让团队成员同步进展、解决问题。在软考阅卷系统开发中，这种高效沟通机制确保了信息的及时传递，使问题能迅速得到处理，提高了开发效率和系统质量，让系统更好地满足软考阅卷的实际需求。</p> <p>在实际的软考阅卷系统开发中，要想有效应对需求的动态变化、快速响应政策调整以及保障开发的执行效率，需要一系列科学合理的机制。接下来，我将详细阐述以用户故事为核心的需求拆分机制、每日站会驱动的团队高效沟通协作机制在其中发挥的重要作用。</p>
主要内容 1000	<p>以用户故事为核心的需求拆分机制是敏捷开发应对动态需求的基础。在软考阅卷系统开发初期，我们就遇到了需求频繁变动的问题。假如沿用传统的需求规格说明书，把"阅卷系统"当作一个整体功能块来写，势必会出现了大量理解偏差——传统整体化描述未明确划分试卷扫描录入、智能阅卷、人工复评等核心模块的职责边界，易使不同执行角色产生认知偏差。例如，开发团队可能将“试卷扫描后自动匹配评分标准”默认为扫描模块的内置功能，而软考办实际期望扫描模块仅负责数据采集，评分标准匹配由独立的评分标准设定模块完成。这种偏差会导致开发阶段模块功能重叠，后期需投入额外人力拆分重构——据类似项目经验，此类问题可能使开发周期延长 15%，且增加 30% 的代码修改工作量。此外，测试团队在整体化需求下难以制定针对性测试用例，无法精准验证各模块功能完整性，可能遗漏“扫描模块页码校正失败后未触发异常提醒”等关键问题，影响系统上线后的稳定性。因此，我们转向用户故事拆分，把复杂需求拆解成"试卷录入""评分规则设定""成绩统计"等小单元：比如"作为阅卷老师，我需要录入试卷基本信息（科目、题号、考生 ID），以便系统定位试卷并分配阅卷任务"，明确角色（阅卷老师）、目标（定位试卷）和价值（分配任务）；对"评分规则"，我们用"作为考务管理员，我需要设定不同题型的评分细则（如客观题自动判分、主观题按采分点给分），以便系统按规则计算成绩"来描述。这套方法推行后，效果立竿见影，考试政策调整时，我们仅需新增一个用户故事，2 周就完成开发，比之前整模块返工缩短了一半时间，需求变更响应速度和开发效率都明显提升。</p> <p>每日站会驱动的团队高效沟通协作机制是保障敏捷开发执行效率的关键，</p>

	<p>在软考阅卷系统的开发过程中，我对此深有体会。系统开发初期，我们 12 人团队分评分算法、试卷存储、权限管理三个模块并行开发，当时依赖微信群和周会同步进度，结果信息传递严重滞后。比如在某个智能解析模块，评分算法组按“关键词匹配权重”开发，而试卷解析组理解的是“语义相似度优先”。这个问题两周后整合时才发现接口逻辑冲突，光返工就耗了 3 人天。我们曾尝试过每日下班前发进度文档，但大家常因赶工拖延，文档要么简略要么滞后，问题暴露平均要等 2-3 天。后来严格推行每日站会机制：每天早上 9 点准时开始，每人严格控制在 3 分钟内说清“昨天完成了什么（如‘完成主观题评分接口联调’）、今天计划做什么（如‘开发试卷异常标记功能’）、遇到什么阻碍（如‘测试环境数据库连接数不足’）”，用 Excel 实时记录阻碍项，会后立即按问题类型拉群。技术难题（如算法冲突）当天 10 点前拉开发+产品专项会，资源瓶颈（如测试环境）直接@运维组协调。推行站会后，团队信息同步周期从周缩短到天，问题平均响应时间从 72 小时压缩到 4 小时，那个曾因环境问题延误的迭代，后续同类模块开发周期缩短了 20%，系统上线后试卷评分准确率达 99.7%，未出现因沟通滞后导致的功能缺陷。</p>
结尾 400 字	<p>通过在软考阅卷系统项目中运用敏捷开发，项目取得了显著成效。系统按时交付上线，不仅实现了高效准确的阅卷功能，极大提升了阅卷效率，还能根据实际需求随时调整与优化，满足了软考组织方不断变化的需求，得到了各方高度认可。</p> <p>在实践中，我对敏捷开发有了更深入的思考。虽然敏捷开发强调快速响应变化，但这也可能带来需求边界模糊的问题。在本项目里，由于过于强调快速响应需求，有时会出现需求频繁变动，导致团队在开发过程中缺乏稳定的目标，增加了开发的复杂度和成本。而且，敏捷开发高度依赖团队成员之间的沟通与协作，如果团队成员之间的沟通不畅或者协作不紧密，就容易出现信息传递误差，影响项目进度和质量。此外，敏捷开发注重短期迭代和交付，可能会让团队过于关注眼前的功能实现，而忽视了系统的长期架构设计和技术积累，不利于系统的可持续发展。所以，在运用敏捷开发时，我们要平衡好快速响应与需求管理、团队协作以及长期规划之间的关系。</p>

12. 论文自查清单

论文科目最多的还是练，关键等我论文押题出来之后，参照押题进行训练。

具体要做的工作如下表所示。在论文备考期间，严格执行完成下表内容。

论文字数是否达标（严格参照字数分配章节）
是否使用 AI 助手用于内容创作（每人 15 篇机会）
是否生成过项目 PRD 用于系统理解企业级项目要解决的问题
是否模拟过考试场景 2 小时限定时间内写作
论文批改 3 次是否已经用过
不要抄范文，用 AI 助手代替抄，否则要翻车