项目计划书

项目名称：在线考试系统

项目成员：苟瑞祥 李轩珂 侯一玮

胡禹晨 陈致江

所属院校：软件学院

时间：2025.2.26

（第二次修改：2025/3/5）

目录

[1 引言 1](#_Toc194480958)

[2 项目概述 2](#_Toc194480959)

[3 项目团队组织 3](#_Toc194480960)

[4 实施计划 3](#_Toc194480961)

[5 模拟预算 4](#_Toc194480962)

[6 测试方案 4](#_Toc194480963)

[7 附录 5](#_Toc194480964)

## 1 引言

###### 1.1 编写目的

本计划书旨在明确在线考试系统的开发目标、实施路径及资源规划，为项目团队提供执行指南，确保在2025年5月前完成系统开发并上线运行

6

###### 1.2 项目背景

响应《教育信息化2.0行动计划》政策导向，解决传统考试模式三大痛点：

​组织效率低下：纸质考试耗费90%人工成本在印刷、监考环节

1

​数据分析滞后：成绩统计平均耗时3工作日/场次，缺乏知识点分析维度

​防作弊成本高：中小机构监控系统部署成本超5万元/年

3

###### 1.3 定义

​OES：Online Examination System 在线考试系统

​JWT：JSON Web Token 无状态身份认证协议

​Shiro：Apache权限控制框架

## 2 项目概述

###### 2.1 项目目标

·技术性能目标

并发能力：30人同时考试响应时间≤1.5秒

系统稳定性：99.9%可用性保障，MySQL主从架构容灾设计

3

安全标准：HTTPS传输加密 + BCrypt密码哈希存储

·用户体验目标

教师端：试题导入效率提升60%（支持Excel批量导入）

考生端：答题界面误操作率低于5%（A/B测试优化）

·功能范围

|  |  |  |
| --- | --- | --- |
| 模块 | 功能清单 | 优先级 |
| 核心功能 | 随机组卷、自动阅卷、错题本 | P0 |
| 扩展功能 | 主观题批阅、AI监考 | P2 |

##### 2.2应交付成果

需完成的软件

* B/S架构pc端检测系统
* 移动端
* 开放API接口

需提交用户的文档

* 《用户操作手册》
* 《API接口文档》

须提交内部的文档

* 《项目计划书》
* 《需求规格说明书》
* 《技术设计方案》
* 《开发报告》
* 《测试报告》

应当提供的服务

* 系统部署与运维支持
* 用户培训与技术咨询。

###### 2.3 项目开发环境

项目开发基于Java 17与Vue.js 3.3构建，采用SpringBoot 3.2框架实现微服务架构，集成Shiro 1.12权限框架及JWT 0.12令牌认证机制构建核心业务模块。开发环境通过Docker Desktop 5.0虚拟化MySQL 8.0集群与Redis 7.0缓存服务，确保本地开发与云端部署环境一致性。代码开发与调试依托IntelliJ IDEA 2025.1完成，其SpringBoot Assistant插件与Database Navigator工具链支持RESTful接口调试与ORM映射验证。

团队协作采用GitCode作为代码托管平台，遵循Epic（考试生命周期管理）-Feature（随机组卷算法）-Task（哈希密码存储实现）三级敏捷拆解机制，需求卡片与Git提交记录（Commit SHA-256）通过标签自动化绑定，实现开发过程全链路审计。代码版本控制严格遵循Git Flow策略，生产环境代码锁定main分支并通过合并请求（Merge Request）准入校验，技术文档通过GitCode Wiki按模块化结构归档，支持Markdown格式实时协同编辑。

测试验证环节配置Postman 13.0进行RESTful接口自动化测试，结合JUnit 5.10框架实现核心业务单元测试覆盖率≥90%。系统部署采用Docker Compose 3.0容器化方案，基于Alpine Linux 3.20镜像封装SpringBoot微服务与Nginx 1.25网关，确保从开发调试到生产发布的二进制一致性。环境验证标准包括：Maven依赖树与pom.xml清单完全匹配、组卷服务在IntelliJ热部署模式下响应时间≤1.5秒、HTTPS证书链通过Qualys SSL Labs A+评级。

## 3 项目团队组织

###### 3.1 人员分工

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | 成员 | 角色 | 核心职责 | 技术专长 | | 苟瑞祥 | 项目经理 | 进度控制与资源协调 | PMP认证，Scrum方法论 | | 李轩珂 | 系统分析师 | 需求文档编写与架构设计 | UML建模，Axure原型设计 | | 侯一玮 | 后端开发 | Shiro权限系统与数据库开发 | SpringBoot, MySQL优化 | | 胡禹晨 | 全栈开发 | Vue前端与考试引擎开发 | Element-UI, WebSocket | | 陈致江 | 测试工程师 | 自动化测试与性能调优 | Selenium, JMeter | |

## 4 实施计划

###### 4.1 风险评估及对策

|  |  |  |  |
| --- | --- | --- | --- |
| ​**需求变更** | 30% | 高 | 建立变更控制委员会（CCB），所有需求变更需提交《变更申请单》并通过三方（产品/技术/用户）联合评审，冻结期前2周禁止新增功能需求 |
| ​**技术瓶颈** | 25% | 中 | 采用技术预研机制（PoC验证），核心模块开发预留20%时间缓冲，每周技术复盘会议同步难点解决方案 |
| ​**进度延误** | 35% | 高 | 采用敏捷迭代开发（2周为一个Sprint），设置关键路径里程碑缓冲期（总工期预留15%弹性时间） |
| ​**人员流动** | 15% | 中 | 建立交叉培训机制（核心模块双人负责制），关键设计文档强制纳入《知识库》并设置版本快照 |
| ​**安全漏洞** | 20% | 极高 | 实施SDL安全开发流程，集成SonarQube代码扫描工具，每月委托第三方进行渗透测试（OWASP TOP 10全覆盖） |
| ​**第三方服务依赖** | 18% | 中 | 对支付接口/短信服务等关键依赖建立备用供应商清单，设计服务降级方案（如本地缓存兜底） |
| ​**合规性风险** | 10% | 高 | 法务团队全程参与需求评审，数据存储方案需通过《个人信息保护法》合规性检查，用户隐私数据加密存储 |
| ​**资源不足** | 25% | 中 | 建立资源监控预警机制（CPU/内存阈值≥80%触发告警），采用弹性云架构（阿里云自动扩容策略） |
| 风险类型 | 概率 | 影响 | 应对策略 |
| 需求变更 | 30% | 高 | 建立变更控制委员会（CCB）机制  6 |
| 技术瓶颈 | 25% | 中 | 每周技术预研会议，预留20%缓冲时间 |

###### 4.2 总体进度计划

阶段一：项目启动与规划（第1-3周）

完成《需求规格说明书》评审

确定技术架构选型（SpringBoot+Vue）

阶段二：需求分析与设计（第4-7周）

输出SRS,SDD文档

完成UI原型设计（Axure高保真原型）

阶段三：核心功能开发（第5-9周）

实现随机组卷算法（权重分配策略）

开发自动阅卷引擎（正则表达式匹配）

阶段四：系统集成测试（第10-11周）

执行压力测试（JMeter模拟100并发）

完成安全渗透测试（SQL注入/XSS防护）

阶段五：部署运行（第12-13周）

Docker容器化部署（Nginx+MySQL集群）

用户培训与试运行（3所合作院校试点）

## 5 模拟预算

###### 5.1 人员成本

|  |  |  |  |
| --- | --- | --- | --- |
| 角色 | 人月 | 单价（万元） | 小计 |
| 开发工程师 | 4 | 1.2 | 4.8 |
| 测试工程师 | 1 | 0.8 | 0.8 |

###### 5.2 设备成本

云服务器：阿里云ECS（4核8G）x3台，0.5万元/月

测试设备：Chromium兼容性测试套件，0.3万元

###### 5.3 总经费预算

合计：开发费用4.8万 + 设备费用3.6万 = ​8.4万元

## **6 测试方案**

###### 6.1 功能测试：Selenium自动化测试用例

用例1：模拟用户登录后发起考试，验证自动跳转与倒计时功能。

用例2：管理员批量导入试题，检查题库数量与格式校验逻辑。

###### 6.2 压力测试：JMeter模拟30人并发提交

场景：30用户同时提交试卷，监测服务器CPU/内存占用及数据库锁表现。

优化：启用数据库连接池（HikariCP）、SQL索引优化。

###### 6.3 体验优化：A/B测试界面交互流程



方案A：传统分页答题卡 vs 方案B：单页滚动答题。

指标：用户平均完成时间、误操作率，选择数据更优方案。

**优化**

**功能裁剪**：优先实现单选/判断题型，暂不考虑复杂题型的自动阅卷

**技术简化**：使用轻量级SQLite替代MySQL用于初期开发

**部署方案**：采用Docker容器化部署，降低环境配置难度

**文档管理**：使用GitWiki维护需求文档和API手册

## **7 附录**

###### **7.1 术语表**

**Shiro：Apache开源安全框架，提供认证、授权、加密功能。**

**JWT：JSON Web Token，用于无状态跨域身份验证。**

###### 7.2 参考文档

SpringBoot官方文档：https://spring.io/projects/spring-boot

Vue.js开发指南：https://vuejs.org/v2/guide/

MySQL 5.7性能调优手册：https://dev.mysql.com/doc/refman/5.7/en/