在线考试系统  
软件设计规格说明书（SDD）

依据 IEEE STD 1016-2009

项目名称：在线考试系统（OES）

小 组： 第5组

组 长： 苟瑞祥（项目经理）

组 员： 李轩珂（系统分析师）

侯一玮（后端）

胡禹晨（全栈）

陈致江（测试）

指导教师： 杨波

**评阅成绩：**

修订日期：2025年3月5日

**在线考试系统（OSE）**

**[摘要]**本文档详细描述了符合教育信息化2.0行动规范的在线考试系统（OES）的软件设计规格说明书，依据IEEE STD 1016-2009标准。项目旨在通过微服务架构构建高并发、高可用、智能化的考试系统，解决教育机构在组卷效率、阅卷准确性和系统稳定性方面的痛点。系统采用Vue.js 3 + TypeScript前端、Spring Boot 3 + Cloud Alibaba服务端、MySQL 8.0 + Redis 7.0数据库，部署于阿里云ECS集群，支持30人并发考试场景，关键操作响应时间≤1.5秒。核心功能包括智能组卷（基于遗传算法）、实时监控、安全防护（JWT+Shiro）、自动阅卷等，满足GDPR数据保护要求，提供多维度成绩分析与可视化。非功能性需求涵盖性能、安全性、可用性、可迁移性等，确保系统稳定高效运行。

**关键词**：在线考试系统、微服务架构、智能组卷、遗传算法、安全防护、JWT、Shiro、Spring Boot、Vue.js、MySQL、Redis、教育信息化、高并发、自动阅卷

目录

[一、引言 3](#_Toc195086542)

[1.1 项目背景 3](#_Toc195086543)

[1.2 设计目标 3](#_Toc195086544)

[二、项目概述 4](#_Toc195086545)

[2.1 系统架构 4](#_Toc195086546)

[2.2 核心组件 4](#_Toc195086547)

[2.3 数据流设计 5](#_Toc195086548)

[三、功能需求 5](#_Toc195086549)

[3.1 组卷管理 5](#_Toc195086550)

[3.2 考试监控 5](#_Toc195086551)

[3.3 安全防护 6](#_Toc195086552)

[3.4 部署要求 6](#_Toc195086553)

[四、功能性需求 6](#_Toc195086554)

[4.1 功能划分 6](#_Toc195086555)

[4.2 功能描述 7](#_Toc195086556)

[4.3 用例图 8](#_Toc195086557)

[4.4 活动图 10](#_Toc195086558)

[4.5 时序图 12](#_Toc195086559)

[五、非功能性需求 12](#_Toc195086560)

[5.1 性能 12](#_Toc195086561)

[5.2 质量属性需求 13](#_Toc195086562)

[5.2.1 安全性 13](#_Toc195086563)

[5.2.2 可用性 13](#_Toc195086564)

[5.2.3 可迁移性 13](#_Toc195086565)

[5.3 其他需求 13](#_Toc195086566)

[5.3.1 特殊需求 13](#_Toc195086567)

[5.3.2 异常需求 14](#_Toc195086568)

[六、接口需求 14](#_Toc195086569)

[6.1 人机交互接口 14](#_Toc195086570)

[6.2 内部接口 14](#_Toc195086571)

[6.3 外部接口 14](#_Toc195086572)

# 一、引言

### 1.1 项目背景

随着教育信息化进程的深入发展，传统纸质考试模式已难以满足大规模、高频次、高效率的测评需求。当前，我国教育信息化基础设施初具规模，数字校园蓬勃发展，资源建设与应用初见成效，但仍面临地区发展不均衡、软件环境不协调、师生信息技术能力差异及网络安全等挑战。为应对这些挑战，构建符合教育信息化2.0行动规范的在线考试系统（OES）显得尤为重要。

在线考试系统不仅能支持培训学习、知识竞赛、企业招聘、期末考试等多种场景，还具有安全稳定、高效智能阅卷、数据分析与反馈等优势，可显著提升考试管理的效率和准确性。微服务架构作为先进的技术支撑，具有易于扩展、独立部署、技术栈灵活、易于维护升级和可测试性强等特点，为在线考试系统的高并发、高可用和智能化管理提供了有力保障。同时，遵循IEEE 1016《软件设计描述标准》，可确保系统设计的规范性和实施的有效性。本项目旨在通过微服务架构，设计并实现一个安全、稳定、高效的在线考试系统，解决教育机构在组卷效率、阅卷准确性和系统稳定性方面的痛点，推动教育教学模式的创新，提升教育质量和效率。

### 1.2 设计目标

系统需达成以下核心指标：  
• **性能基准**：支持30人并发考试场景，关键操作响应时间≤1.5秒，MySQL主从架构确保99.9%服务可用性  
• **效率提升**：通过Excel模板实现试题批量导入，较传统逐条录入方式效率提升60%  
• **智能特性**：基于遗传算法的自动组卷引擎，支持多维度约束条件（知识点覆盖、难度系数、题型分布）的智能优化  
• **安全合规**：采用JWT+Shiro实现RBAC权限控制，满足GDPR数据保护要求，通过MyBatis预编译防御SQL注入

#### 1.3 技术选型

| **层级** | **技术栈** | **选型依据** |
| --- | --- | --- |
| **前端** | Vue.js 3 + TypeScript | 渐进式框架支持SPA，TypeScript增强类型安全 |
| **服务端** | Spring Boot 3 + Cloud Alibaba | 微服务架构，集成Nacos服务发现、Sentinel限流 |
| **数据库** | MySQL 8.0 + Redis 7.0 | InnoDB引擎事务支持，Redis实现会话管理与热点缓存 |
| **基础设施** | 阿里云ECS集群 + Docker | 容器化部署提升资源利用率，云原生架构支持弹性伸缩 |

# 二、项目概述

### 2.1 系统架构

### 2.2 核心组件

1. **智能组卷引擎**

**算法核心**：随机组卷（基础实现）+ 遗传算法（规划中）

，适应度函数综合考虑知识点覆盖率、难度标准差、题型比例等约束

**执行流程**: **graph TD**

|  |
| --- |
| * + **A[输入约束条件] --> B[加载Redis试题池]** |
| * + **B --> C[初始化种群]** |
| * + **C --> D[计算适应度]** |
| * + **D --> E{终止条件?}** |
| * + **E -->|否| F[选择/交叉/变异]** |
| * + **F --> D** |
| * + **E -->|是| G[输出最优试卷]自动阅卷系统** |

**客观题处理**：基于正则表达式匹配答案，支持数学公式LaTeX格式校验

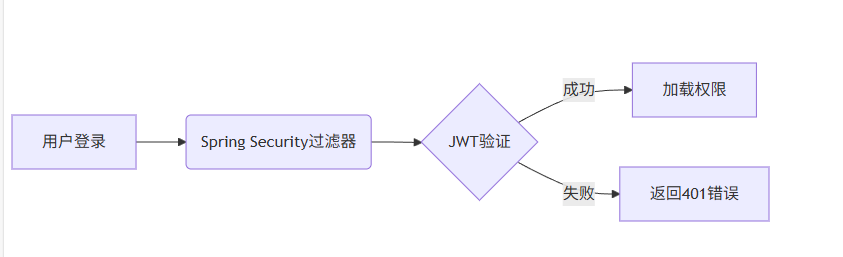
**主观题流程**：采用"初评+复评"双盲评审机制，Kafka消息队列实现任务分发

1. **实时监控服务**

**采集维度**：窗口焦点变化、页面停留时间、异常退出行为

### ****技术实现****：WebSocket全双工通信，基础行为日志记录（窗口切换次数/IP检测）

### 2.3 数据流设计



# 三、功能需求

### 3.1 组卷管理

• **需求描述**：支持多维度约束的自动组卷，需实现知识点标签匹配、难度系数控制（±0.2误差）、题型比例分配（精确到5%）  
• **性能要求**：生成包含50题的试卷≤3秒，并发组卷请求处理量≥100次/分钟  
• **扩展需求**：支持手动调整试题顺序，提供试题替换建议（基于相似度算法）

### 3.2 考试监控

• **实时监控**：需捕获考生异常行为，包括：

* 窗口切换次数＞5次/小时
* 页面停留时间＜答题最小预估时间×50%
* 多IP同时登录（基于IP地址库校验）  
  • **数据留存**：监控日志需保存180天，支持按考生ID、考试时间范围检索

### 3.3 安全防护

• **认证机制**：Spring Security 6.2 + JWT动态密钥管理

• **权限控制**：

* **管理员**：试卷管理、用户管理、系统监控
* **教师**：试题录入、组卷配置、成绩分析
* **学生**：参加考试、查看历史成绩  
  • **审计要求**：所有敏感操作（试卷删除、成绩修改）需记录操作日志，包含：
* 操作人用户ID
* 操作时间（精确到毫秒）
* 影响数据范围

### 3.4 部署要求

• **容器化配置**：（dockerfile）

|  |  |
| --- | --- |
|  | # Spring Boot服务镜像 |
|  | FROM eclipse-temurin:17-jdk |
|  | VOLUME /tmp |
|  | COPY target/\*.jar app.jar |
|  | ENTRYPOINT ["java","-jar","/app.jar","--spring.profiles.active=prod"] |

• **集群配置**：

* MySQL主从延迟＜3秒，通过Percona Toolkit监控复制状态
* Redis集群采用3主3从架构，内存预分配策略避免动态扩容
* Nginx配置连接超时时间30秒，最大客户端数2000

# 四、功能性需求

### 4.1 功能划分

系统核心功能模块划分如下：  
**用户管理模块**

* **功能范围**：用户注册（邮箱/手机号验证）、多角色登录（Admin/Teacher/Student）、权限动态分配（基于RBAC模型）
* **技术实现**：使用Spring Security实现认证流程、JWT令牌实现无状态会话

**试题管理模块**

* **功能范围**：Excel模板化批量导入（支持题型：单选、多选、填空、判断题）、试题标签管理（知识点分类、难度等级、章节归属）、试题版本控制（历史修改记录追溯）
* **技术实现**：Apache POI解析Excel文件、Redis缓存试题元数据（标签索引加速查询）

**组卷与考试模块**

* **功能范围**：智能组卷（遗传算法优化试题组合）、考试过程控制（倒计时、自动保存、防作弊监控）、试卷版本发布与撤回
* **技术实现**：遗传算法适应度函数（权重匹配度 + 知识点覆盖率）、WebSocket实时推送考试事件（如剩余时间提醒）

**阅卷与成绩模块**

* **功能范围**：客观题自动评分（精确匹配与模糊匹配策略）、主观题教师批阅（评分标准模板化）、成绩统计报表（多维分析：班级排名、知识点薄弱点）
* **技术实现**：正则表达式处理填空题容错（如大小写、空格忽略）、ECharts可视化成绩分布

**系统监控模块**

* **功能范围**：实时考试监控大屏（在线人数、异常行为告警）、日志审计（操作日志、系统错误日志）、资源使用监控（CPU、内存、数据库连接池状态）
* **技术实现**：Prometheus + Grafana监控体系、ELK（Elasticsearch, Logstash, Kibana）日志分析

### 4.2 功能描述

**智能组卷（优先级：高）**

* **输入**：必选参数（知识点分布、难度系数）、可选参数（题型数量限制）
* **处理流程**：加载试题池、初始化种群、迭代优化（交叉/变异/适应度评估）、输出最优试卷
* **输出**：试卷ID、试题列表（JSON结构）
* **约束**：遗传算法迭代次数≤100次、单次组卷耗时≤30秒

**在线考试（优先级：最高）**

* **核心流程**：身份认证、试卷加载、答题过程（实时保存、防作弊监控）、交卷处理（正常提交、超时强制提交、异常处理）

**自动阅卷（优先级：高）**

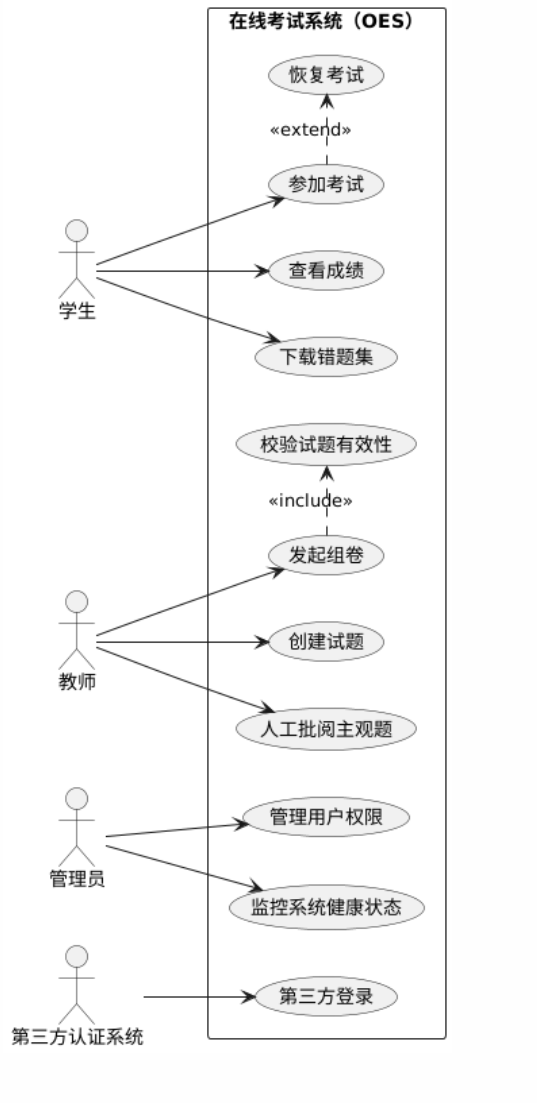
* **评分规则**：单选题/判断题（完全一致）、多选题（全对得满分，部分选对按比例得分）、填空题（正则表达式模糊匹配）
* **技术实现**：MyBatis动态SQL批量比对答案、Redis缓存标准答案

**系统监控（优先级：中）**

* **监控指标**：微服务健康状态、数据库负载、缓存命中率
* **告警策略**：阈值触发（CPU使用率>80%持续5分钟）、通知方式（邮件、钉钉机器人）

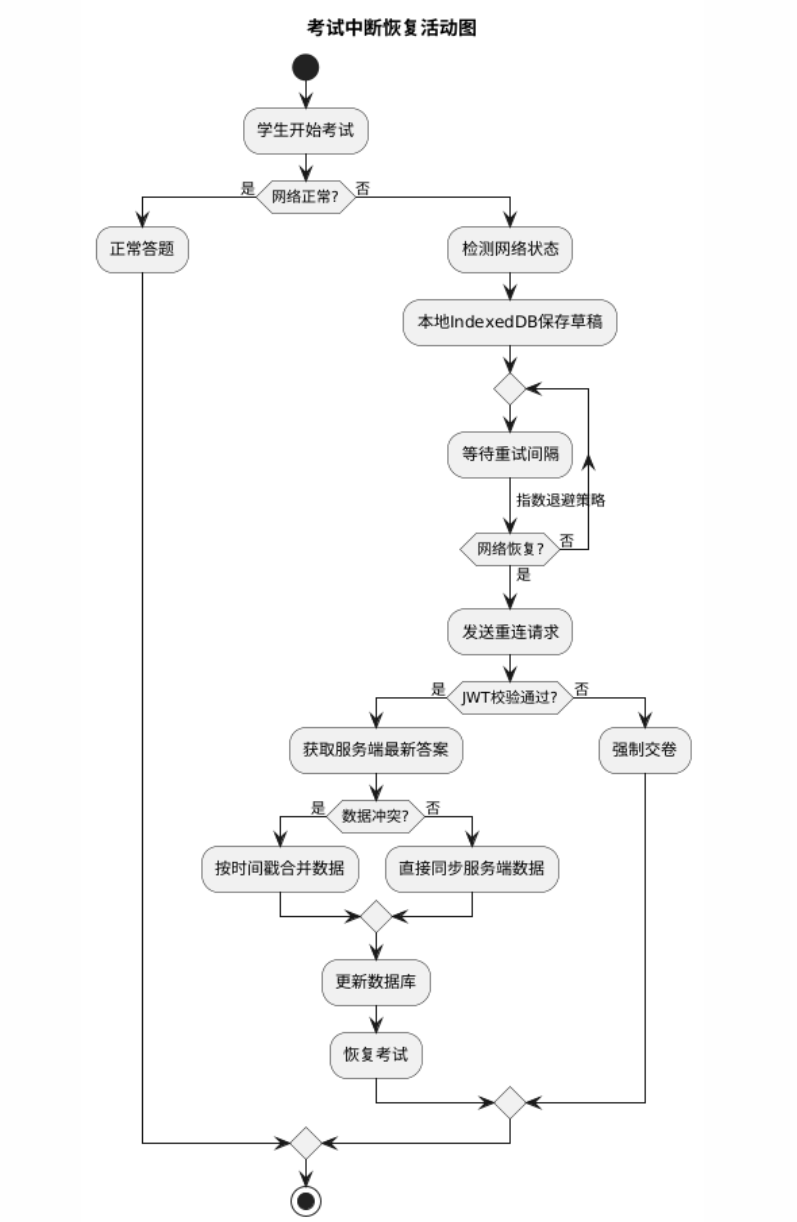
### 4.3 用例图

**参与者与用例详解**：

* **学生**：参加考试、查看成绩、下载错题集
* **教师**：创建试题、人工批阅
* **管理员**：权限管理、数据备份
* 

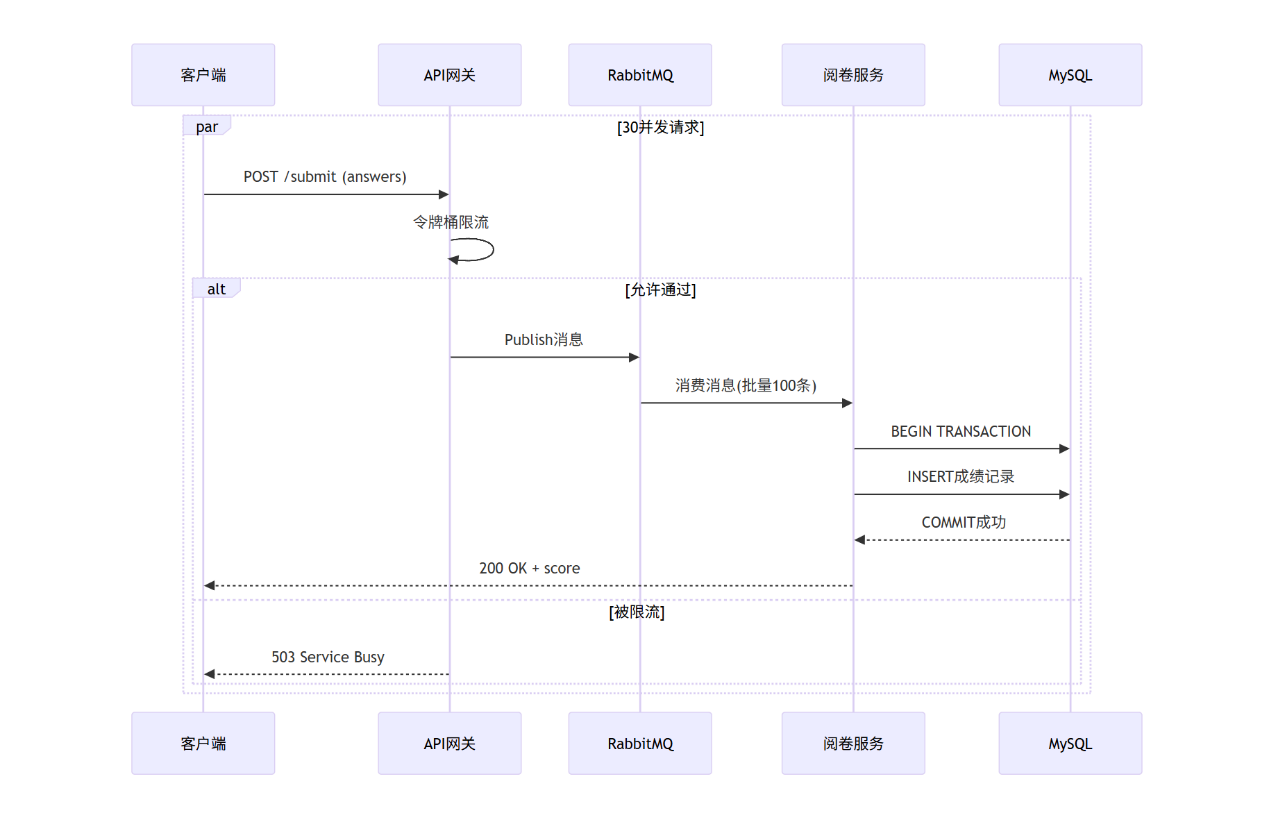
### 4.4 活动图

**核心流程：考试中断恢复**

* **学生端活动**：检测网络状态、本地暂存答案、定时重连
* **服务端活动**：校验令牌有效性、对比答案版本、冲突解决、数据同步
* 

### 4.5 时序图

**场景：高并发考试提交**

* **客户端**：用户点击提交按钮、前端构造请求体
* **API网关**：鉴权、限流
* **阅卷微服务**：异步消息队列接收请求、消费者进程批量处理
* **数据库**：MyBatis批处理插入成绩记录、事务管理
* **响应返回**：成功（HTTP 200）、失败（HTTP 503）
* 

# 五、非功能性需求

### 5.1 性能

| **场景** | **指标** | **阈值** | **测试方法** |
| --- | --- | --- | --- |
| 并发考试 | 平均响应时间 | 50并发/TP99≤2秒（需优化） | JMeter模拟30线程循环提交 |
| 试题导入 | 数据解析速度 | 1000题≤30秒 | 监控POI解析耗时 |
| 自动阅卷 | 单次评分延迟 | ≤200ms | 统计MyBatis批处理执行时间 |
| 故障恢复 | MySQL主从切换时间 | ≤30秒 | 人工触发主库宕机，记录恢复时间 |

### 5.2 质量属性需求

### ****5.2.1 安全性****

* **数据安全**：

1. BCrypt密码哈希存储（强制强度≥10）

2. JWT密钥每日轮换（通过Redis管理）

3. 设备指纹绑定（IP+UserAgent哈希）

* **审计追踪**：日志脱敏（身份证号、手机号部分掩码）、操作追溯（记录关键操作IP、设备指纹）

### ****5.2.2 可用性****

* **容错设计**：服务降级（Redis不可用时切换至MySQL）、熔断机制（Hystrix控制微服务依赖超时）
* **灾备方案**：跨地域备份（阿里云多可用区部署）、冷备数据（每周全量备份至异地OSS）

### ****5.2.3 可迁移性****

* **环境适配**：JDK版本（兼容OpenJDK 11~17）、数据库（支持MySQL 5.7/8.0、MariaDB 10.3+）
* **迁移工具**：提供数据导出脚本（CSV、JSON）、容器化部署（Kubernetes Helm Chart打包）

### 5.3 其他需求

### ****5.3.1 特殊需求****

* **国际化**：前端多语言支持（中/英文切换）、时区适配（考试时间按用户本地时区显示）
* **法律合规**：GDPR数据隐私保护（用户数据导出/删除接口）、等保三级要求（日志留存≥180天）

### ****5.3.2 异常需求****

* **恶意攻击防护**：IP黑名单（Fail2ban自动封禁高频错误请求）、验证码机制（密码错误≥3次触发图形验证码）
* **数据一致性**：最终一致性（允许阅卷结果延迟1分钟同步）、补偿事务（消息队列消费失败时重试3次 + 人工干预告警）

# 六、接口需求

### 6.1 人机交互接口

* **UI规范**：颜色规范（错误提示红色#FF4444、成功绿色#00C851）、交互反馈（按钮防重复点击）
* **辅助功能**：键盘导航支持（Tab键切换焦点）、屏幕阅读器兼容（ARIA标签）

### 6.2 内部接口

* **微服务间调用**：服务发现（Nacos注册中心）
* **示例接口**：

|  |  |
| --- | --- |
|  | * POST /internal/paper/generate |
|  | * Content-Type: application/json |
|  | * { |
|  | * "knowledgePoints": ["OS", "Network"], |
|  | * "difficulty": 0.75 |
|  | * } |

* **缓存更新**：失效策略（试题修改时发布Redis PUBLISH）、监听处理（组卷服务订阅频道并刷新本地缓存）

### 6.3 外部接口

* **支付网关**：协议（支付宝开放平台API V2）、加密方式（RSA2签名）、回调验证（异步通知验签 + 订单状态同步）
* **单点登录（SSO）**：协议（SAML 2.0）、元数据交换（IDP与SP双向证书校验）
* **短信服务**：供应商（阿里云短信服务）、频率限制（同一手机号每日≤10条）、模板示例：

### 7 附录

### 7.1 术语表

| 术语 | 定义 |
| --- | --- |
| Spring Security | 企业级认证授权框架，支持OAuth2/JWT |
| BCrypt | 自适应哈希算法，抗彩虹表攻击 |