



奖项识别系统设计与 实现

月下调试人





目录

CONTENTS

01 评审要点与核心回应

系统与数据一体化设计 02

03 改进建模图解读

团队分工与交付节奏 04





01

评审要点与问题回应





评审维度与得分权重

全面把握评审标准，精准呈现项目价值



功能模块层次设计 (20%)

模块划分合理，层次清晰



PPT质量 (10%)

结构清晰、美观易读



接口设计 (15%)

RESTful风格，统一响应格式



系统安全性和健壮性 (15%)

加密、权限、备份、监控



存在问题和建议 (5%)

识别风险，提出改进

ER图和表结构设计
(25%)

表关系明确，索引合理



演讲 (10%)

表达清晰，重点突出



上节课问题回应：为什么加入LLM？

LLM超越OCR的语义价值

从“看见”到“理解”，是奖项智能识别的核心跃迁

OCR (光学字符识别)



- 核心能力: 提取图像中的文字信息。
- 局限性: 无法理解语义，无法判断“英语竞赛二等奖”与“全国大学生英语大赛二等奖”的等价关系。

LLM (大型语言模型)



- 核心能力: 进行语义理解与标准化。
- 关键作用: 匹配奖项名称、智能分类评分、容错处理，是确保高准确率与自动化的核心组件。



02

系统与数据一体化设 计



分层微服务架构全景

通过分层隔离故障域，支持独立扩容，为高并发与可维护性奠定基础



前端展示层

Vue.js 构建用户界面



API 网关层

Nginx + Lua 负责路由与限流



业务服务层

上传、识别、审核、通知等微服务



AI 服务层 (核心)

聚合 CLIP (预判) → OCR (提取) → LLM (理解) → 向量检索 (查重)



数据存储层

MySQL, ES, Redis, MinIO 各司其职

功能模块拆解与协作流程

服务间通过消息队列解耦，支持失败重试与流量削峰，保障系统韧性

上传管理服务



- 批量上传文件
- 格式与大小校验
- 存储至 MinIO

识别处理服务



- CLIP 预判 → OCR 提取
- 向量转换 → LLM 分类
- 查重 → 入库

审核管理服务



- 自动通过策略
- 人工审核与批量处理
- 审核记录追踪

RESTful接口统一规范

遵循RESTful风格，统一响应格式，降低沟通成本，为第三方集成预留标准入口

POST

/api/v1/upload

文件上传

GET

/api/v1/task/{taskId}

任务查询

PUT

/api/v1/audit/{taskId}

审核操作

统一响应格式

```
{ "code": 200, "msg": "success", "data": {...}, "requestId": "..." }
```

安全与健壮性六重保障

构筑可信运行环境，保障数据安全与业务连续性



身份认证

JWT + SSO 无状态登录



权限控制

RBAC 模型，细粒度授权



数据加密

AES-256 & HTTPS



备份恢复

全量+增量备份，月度演练



系统监控

Prometheus + Grafana



等保三级预审

通过权威机构安全认证

数据库ER图与核心表结构

表关系明确，通过复合索引、覆盖索引、分区策略优化千万级记录查询性能。

核心表清单

- users: 用户信息
- awards: 奖项规则与分类
- tasks: 任务流程与状态
- audit_logs: 审核记录追踪
- statistics_summary: 统计摘要与积分

索引与安全策略

索引优化

采用复合索引、覆盖索引，并对大表进行分区，显著提升查询效率。

安全策略

对敏感字段（如身份证号）实施字段级加密，兼顾性能与安全。

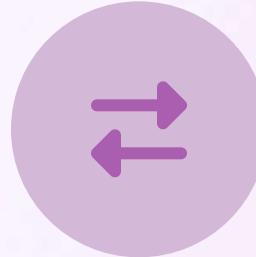
性能与扩展性设计策略

策略组合保证系统在奖项高峰期仍保持低延迟与高可用



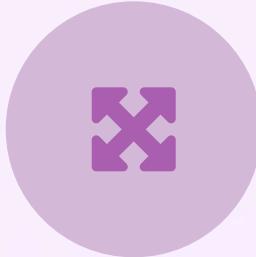
三级缓存

Redis缓存会话、热点数据及队列状态，降低数据库压力。



异步处理

RabbitMQ异步化处理识别任务，实现流量削峰，提升响应速度。



水平扩展

微服务无状态化，数据库读写分离+分片，K8s自动扩容。



03

改进建模图解读



改进后的类图：

数据流图(DFD)三级细化

逐级细化，精确把握数据走向，暴露潜在瓶颈，为性能优化提供依据



顶层图 (Context)

系统黑盒，展示与外部交互



一级细化 (Level 1)

拆解为上传、识别、审核、统计



二级细化 (Level 2)

聚焦识别：OCR → 向量 → LLM → 匹配

泳道图：描绘任务处理的完整旅程

清晰区分人机职责，突出审核环节的人工兜底作用，强化系统可信性





04

团队分工与交付节奏



四周敏捷开发里程碑

每周设置可演示的交付物，采用每日站会+燃尽图跟踪，确保评审前三就绪

1

第一阶段

基础架构搭建、核心功能开发

2

第二阶段

AI模型集成、接口联调

3

第三阶段

功能完善、性能测试

4

第四阶段

系统测试、部署上线

角色职责与人力配置

通过RACI表明确职责，减少沟通开销，体现团队协作的规范化

核心角色

曾诗皓 (组长): 项目管理、进度协调

韩冰晨 (产品): 需求分析、任务分配、PPT制作

张宝仁 (开发组长): 前后端开发、技术架构

支持角色

池博洋: 博客撰写、文档整理

黄俊平、何振坤、王盛明: 测试、辅助开发

工作流程

任务拆分 → 开发组实现 → 测试验收 → 文档/博客/PPT

贡献度量化

综合评定贡献

贡献比例

张宝仁 (开发): 30%



韩冰晨 : 14%



池博洋 : 12%



其他成员 (文档/测试/开发): 各 11%

曾诗皓 : 11%



王盛明: 11%



何振坤 : 11%



黄俊平 : 11%





感谢观看

