**PopQuiz（PQ)项目规划书**

目录

[一、项目概述 1](#_Toc24128)

[（一）项目背景 1](#_Toc14755)

[（二）项目目标 1](#_Toc20550)

[（三）项目范围 1](#_Toc20204)

[二、 团队分工 2](#_Toc5240)

[（一） 项目负责人：刘建顺 2](#_Toc9162)

[（一）后端开发：念雪南、刘建顺 2](#_Toc10406)

[（二）前端开发：李正鹏、郭钰 2](#_Toc18829)

[（三）AI 模块开发：念雪南、李正鹏、刘建顺 2](#_Toc16530)

[（四）测试与运维：郭钰 2](#_Toc21956)

[三、时间计划（总周期 13天，7.10——7.23） 3](#_Toc20770)

[（一）7.10-7.11：需求分析与设计 3](#_Toc28955)

[（二）7.12-7.15：核心功能开发 3](#_Toc21029)

[（三）7.16-7.20：功能完善与集成 3](#_Toc18075)

[（四）7.21-7.23：系统测试与优化 3](#_Toc21291)

[（五）7.24-7.25：部署上线与总结 3](#_Toc7569)

[四、里程碑 4](#_Toc9289)

[五、风险管理 4](#_Toc31705)

[（一）风险识别 4](#_Toc6581)

[（二）应对措施 4](#_Toc23400)

[六、资源需求 4](#_Toc27049)

[七、项目验收标准 5](#_Toc5451)

## **一、项目概述**

### **（一）项目背景**

在各类知识分享场景中，演讲者难以确认听众对内容的理解程度，听众也难以及时反馈不同意见。PopQuiz（PQ）项目旨在利用 AI 大模型提供智能随堂测验，解决上述痛点，提升知识分享的效果和质量。

### **（二）项目目标**

开发集输入收集、处理、输出与用户交互于一体的 PQ 系统，通过网页应用，实现实时互动检测、题目生成、答题反馈、数据统计分析等功能，保障用户数据隐私，满足演讲者、听众、组织者三类用户的需求。

### **（三）项目范围**

涵盖演讲前、演讲中、演讲结束、事后四个环节的全流程功能，包括输入收集（文本、PowerPoint、PDF、音频、视频）、内容处理（题目生成等）、输出与用户交互（各用户视角功能、公共需求等），以及非功能需求（用户体验、数据隐私、性能、兼容性）的实现。

## **团队分工**

### **项目负责人：刘建顺**

1. 负责项目整体规划、进度把控、团队协调。

### **（一）后端开发：念雪南、刘建顺**

1. 进行后端架构设计，数据库设计，处理数据存储与交互。
2. 实现用户管理功能（注册、登录、个人信息管理）。
3. 开发演讲活动创建、信息存储、多活动并发处理等功能。

### **（二）前端开发：李正鹏、郭钰**

1. 负责网页应用和微信小程序的界面设计与开发。
2. 实现用户交互功能，如演讲者创建活动、发送题目，听众答题、反馈等。
3. 完成数据可视化展示（图表等）和徽章体系的前端呈现。
4. 保证前端与后端的顺畅对接，优化用户体验。

### **（三）AI 模块开发：念雪南、李正鹏、刘建顺**

1. 负责 AI 大模型的集成与调用，实现多模态内容（语音、PPT、视频等）到文字的转换。
2. 开发题目生成功能，确保在 10 秒内生成符合要求的个性化选择题。
3. 建立质量提升闭环，优化题目质量，解决题目肤浅、文不对题等问题。
4. 处理 AI 分析相关的逻辑，如根据演讲内容生成题目等。

### **（四）测试与运维：郭钰**

1. 制定测试计划，进行独立模块测试、集成测试，涵盖功能、性能、兼容性等方面。
2. 重点测试多活动同时进行的场景，确保系统稳定运行。
3. 负责项目部署、服务器维护、数据备份等运维工作。
4. 收集用户反馈，协助团队进行系统优化。

## **三、时间计划（总周期 13天，7.10——7.23）**

### **（一）7.10-7.11：需求分析与设计**

1. 团队成员深入研读需求分析说明书，明确各项需求。
2. 完成系统架构设计、数据库设计、UI/UX 设计。
3. 制定详细的开发计划和测试计划，确定技术选型。

### **（二）7.12-7.15：核心功能开发**

1. 后端：完成用户管理、演讲活动管理、数据存储等核心接口开发。
2. 前端：开发基础界面，实现演讲前、演讲中、演讲结束、事后各环节的用户交互原型。
3. AI 模块：实现多模态内容转文字功能，初步完成题目生成功能。
4. 测试：同步进行单元测试，及时发现并反馈问题。

### **（三）7.16-7.20：功能完善与集成**

1. 后端：完善接口，实现多活动并发处理、数据统计分析等功能。
2. 前端：完成所有界面开发，实现数据可视化、徽章体系、反馈提交等功能，与后端完成集成。
3. AI 模块：优化题目生成效率和质量，建立质量提升闭环。
4. 测试：进行模块集成测试，重点测试各模块间的交互。

### **（四）7.21-7.23：系统测试与优化**

1. 全面开展功能测试、性能测试、兼容性测试，确保满足需求。
2. 针对测试中发现的问题，团队协作进行修改优化。
3. 进行用户体验优化，如引导页面、提醒方式等。

### **（五）7.24-7.25：部署上线与总结**

1. 进行上线前最终测试，确保系统稳定运行。
2. 进行项目演示、项目总结，成功结项。

## **四、里程碑**

1. 7.11日：完成设计文档，确定技术方案，团队分工明确。
2. 7.15：核心功能开发完成，实现基本的演讲前、演讲中功能流程。
3. 7.20：所有功能开发完成，模块集成完毕，能进行完整的功能演示。
4. 7.22：测试通过，系统性能、兼容性等满足需求。
5. 7.24：系统成功演示，项目总结完成。

## **五、风险管理**

### **（一）风险识别**

1. AI 题目生成效率不达标，无法在 10 秒内完成。
2. 多活动同时进行时，系统出现并发处理问题。
3. 用户体验不符合预期，如操作复杂、响应缓慢。
4. 数据隐私保护存在漏洞。

### **（二）应对措施**

1. 对于 AI 题目生成效率问题，提前进行技术调研和测试，选择高效的 AI 模型，优化算法。
2. 针对多活动并发处理，在设计阶段采用合适的架构（如分布式处理），加强集成测试，模拟高并发场景进行压力测试。
3. 为提升用户体验，在开发过程中定期进行内部试用，收集意见及时调整，邀请少量外部用户进行 beta 测试，根据反馈优化界面和流程。
4. 关于数据隐私，严格按照需求实现匿名选项和数据加密，在测试阶段重点检查隐私保护功能，确保符合相关规范。

## **六、资源需求**

1. 开发工具：前端（Vue等）、后端（Java等）、数据库（MySQL等）、AI 模型接口（如 GPT 等）。
2. 硬件设备：开发电脑、服务器（用于部署）。
3. 协作工具：Git（版本控制）、腾讯会议（沟通）、微信群聊。
4. 测试资源：不同型号的手机（测试小程序兼容性）、不同浏览器（测试网页兼容性）。

## **七、项目验收标准**

1. 功能完整性：所有需求分析中提到的功能模块均实现，如输入收集支持多种格式，题目生成符合要求，各用户视角功能正常使用。
2. 性能指标：输入处理快速，题目生成在 10 秒内完成，系统响应及时，多活动并发运行稳定。
3. 兼容性：支持主流浏览器和微信版本，不同设备上运行正常。
4. 用户体验：引导页面清晰，注册登录流程简便，提醒方式友好，数据展示直观。
5. 数据隐私：匿名功能有效，数据安全得到保障。

通过以上项目规划，确保 PopQuiz（PQ）项目能够有序推进，按时保质完成，满足用户需求。