**技术原型迭代计划**

　　　　　　　　　　　　　制定日期：2023.6.26

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 2 | 项目名称 | “Testland”--在线代码测评平台 |
| 迭代名称 | 技术原型迭代 | 计划起止日期 | 6.27-7.5 |
| **任务、进度安排和人员分配**  **任务1：前端界面和功能**   * 任务：根据项目要求和用户需求，设计前端页面并进行美化和优化；确保能与后端正常通信，各页面功能正常运行。 * 进度安排： * 6.27-7.1进行界面美化和样式调整。 * 6.27-7.2：进行接口文档编写。 * 7.1-7.2同后端进行跨域通信。 * 7.2至迭代结束优化完善页面设计。 * 人员分配：@胡彤   **任务2：后端功能和接口**   * 任务：根据需求完成数据库设计、接口设计与开发、业务逻辑开发。 * 进度安排： * 6.27-7.1：进行接口开发和功能实现。 * 7.1-7.2：进行单元测试和修复，同后端进行跨域通信。 * 6.27-7.2：进行接口文档编写。 * 7.2至迭代结束：进行功能完善及测试。 * 人员分配：@杨菡雪   **任务3：搭建评测模块和消息队列**   * 任务：搭建评测模块和消息队列，确保能够正确接收和处理测评信息，并将结果返回给后端。 * 进度安排： * 6.27-7.2 并行进行评测模块和消息队列的搭建。 * 7.2-7.3 完成模块集成并接入后端。 * 7.3 至迭代结束：功能测试。 * 人员分配：@张奕涵@全雨乐   **任务4：迭代总结**   * 任务：对本次迭代成果进行测试，完成迭代评估报告 * 进度安排：本迭代后期完成 * 人员分配：开发小组成员 |  |  |  |
| **预期成果**   * 主体功能的前后端代码 * 支持用户内容安全性过滤的评测代码 * 迭代评估报告 * 已有代码的单元测试代码和报告 |  |  |  |
| **主要的风险和应对方案**  **技术风险：**   1. 技术选型风险：选取的技术可能无法满足项目需求或者存在严重的性能问题。为了缓解这个风险，我们可以在选择技术时进行充分的调研和评估，包括对比不同技术的优缺点和实际应用案例。同时，可以进行一些小规模的原型验证，以确定技术的可行性和适用性。 2. 集成风险：由于项目中需要整合多个组件或服务，可能存在集成过程中的兼容性、接口定义不一致等问题。为了降低集成风险，可以在开发过程中尽早进行项目集成，确保各个组件能够正确地协同工作。   **进度风险：**   1. 开发任务过于复杂或估计不准确：在项目进行过程中，可能会发现某些开发任务比预期的要复杂，导致进度延迟。为了应对这个风险，可以在项目开始前进行充分的需求分析和任务拆分，确保对开发工作有清晰的认识。另外，及时调整资源和重新安排工作优先级也是应对进度风险的重要手段。 2. 人员变动：由于各种原因，开发人员及开发环境可能会发生变动，例如请假、休息日等。为了缓解这一风险，可以在项目开始时建立良好的知识共享和团队协作机制，确保团队成员之间有足够的交流和合作。此外，应当建立文档和代码库的规范化管理。 3. 第三方依赖问题：项目中可能涉及使用第三方的组件、服务或库，而这些第三方可能出现不稳定、不可用或存在安全漏洞等问题。为了应对这个风险，可以在选择第三方依赖时进行充分的评估和测试，选择稳定可靠的版本，并及时更新和修复存在的问题。同时，建立灵活的组件替换策略，可以在必要时快速切换到其他可用的解决方案。   **质量风险：**   1. 缺乏充分的测试覆盖：如果测试不够充分，可能会导致潜在的缺陷未被发现，影响系统的稳定性和可靠性。为了降低质量风险，可以制定详细的测试计划，包括单元测试、集成测试和系统测试等，并建立自动化测试框架，确保代码的质量和功能的完整性。 2. 代码质量和规范性问题：如果开发过程中忽视了代码质量和规范性，可能会导致代码难以维护和扩展，增加后续开发的风险。为了应对这个风险，可以制定代码开发规范和最佳实践，并进行代码审查和静态代码分析等，确保代码的质量和可维护性。   **管理风险：**   1. 沟通和协调风险：如果团队成员之间沟通不畅或协调不力，可能导致迭代计划不能按时完成。为了应对这个风险，可以建立有效的沟通渠道和协作机制，明确责任和任务分工，并通过每日立会定期进行项目进展和问题沟通会议。   **安全风险：**   1. 用户信息安全问题：如果在设计和实现过程中未考虑用户信息安全，可能导致用户数据泄露或被非法使用。为了减轻安全风险，需要采取适当的安全防护措施，例如数据加密、访问控制和安全审计等。 2. 漏洞和攻击风险：由于系统存在漏洞或未经授权的访问，可能会导致系统遭受恶意攻击，例如网络攻击、跨站脚本攻击等。为了应对这个风险，需要进行安全测试和漏洞扫描，并及时修复发现的安全问题。 |  |  |  |