

【豆包青训营大项目】 前端监控系统项目提交文档

一、项目介绍

本项目是一个完整的前端监控系统，包含 SDK、服务端和监控平台三个主要部分。系统能够实现前端应用的错误监控、性能监控、用户行为追踪等功能，帮助开发团队及时发现和解决问题。

项目服务地址：远程部署在 localhost 环境下

- 监控后台：<http://localhost:3000>
- 测试页面：<http://localhost:3030>
- 服务端：<http://localhost:8001>

项目地址：https://github.com/PDGGK/MarsCode_Winter_project.git （权限设置为 public）

二、项目分工

好的团队协作可以帮助项目更好地完成。以下是我们的分工情况：

团队成员	主要贡献
戴子涵	负责 SDK 开发，包括错误监控、性能监控等核心功能的实现 负责服务端开发，包括日志收集、数据统计分析等功能，负责监控平台前端开发，包括数据可视化、用户界面设计等
倪嘉豪	负责项目文档编写，包括技术文档、使用说明等，参与功能测试和 Bug 修复工作，负责监控平台前端开发，包括数据可视化、用户界面设计等
芮婧	负责项目演示视频制作，参与用户界面测试，协助编写测试用例，负责服务端开发
梁缘	负责项目质量测试，编写自动化测试脚本，参与文档校对和完善，负责 SDK 开发

三、项目实施

3.1 技术选型与相关开发文档

本项目采用了现代化的技术栈和架构设计：

1. 前端监控 SDK：
 - 使用 TypeScript 开发，提供类型安全
 - 采用插件化架构，支持功能按需引入
 - 支持多种上报方式（Beacon、XHR、Image）
 - 内置错误监控、性能监控等核心插件
2. 服务端：
 - Node.js + Express 框架
 - MySQL 数据库 + Prisma ORM
 - 支持日志分析和数据统计
3. 监控平台：
 - Vue3 + TypeScript
 - Element Plus UI 组件库
 - ECharts 数据可视化

3.2 架构设计

系统采用分层架构设计，主要包括：

1. 数据采集层（SDK）：
 - 错误监控：捕获代码错误、资源加载错误、Promise 错误等
 - 性能监控：记录页面加载时间、API 响应时间等
 - 用户行为：追踪用户点击、路由变化等
2. 数据处理层（Server）：
 - 日志接收：处理来自 SDK 的上报数据
 - 数据存储：将数据写入 MySQL 数据库
 - 数据分析：统计分析各类监控指标
3. 数据展示层（Platform）：
 - 实时监控：展示实时错误和性能数据
 - 统计分析：提供多维度的数据分析
 - 可视化图表：直观展示监控结果

3.3 项目代码介绍

项目采用 monorepo 方式管理，主要包含以下目录：



四、测试结果

功能测试

- 1. SDK 功能测试：
 - 错误监控：成功捕获各类错误并上报
 - 性能监控：准确记录性能指标
 - 用户行为：正确追踪用户操作
- 2. 服务端功能测试：
 - 日志接收：正确接收和处理上报数据
 - 数据统计：准确计算各项统计指标
 - API 接口：正常响应前端请求
- 3. 监控平台功能测试：
 - 数据展示：正确显示监控数据
 - 筛选功能：支持多维度数据筛选
 - 图表展示：正确渲染统计图表

性能测试

- 1. SDK 性能：
 - 资源占用小，不影响页面性能
 - 上报机制优化，避免频繁请求

2. 服务端性能：

- 支持高并发数据处理
- 数据库查询优化

3. 监控平台性能：

- 页面加载速度优化
- 大数据量展示优化

五、Demo 演示视频（必填）

演示视频

视频中展示了以下主要功能：

1. 系统整体架构介绍

2. SDK 功能演示

- 错误监控
- 性能监控
- 用户行为追踪

3. 监控平台功能展示

- 数据统计和分析
- 可视化图表
- 筛选和查询功能

六、项目总结与反思

1. 目前仍存在的问题

- 大数据量下的性能优化需要进一步改进
- 某些特殊场景下的错误捕获还不够完善
- 数据分析功能还可以更加丰富

2. 已识别出的优化项

- 添加更多类型的错误监控
- 优化数据存储和查询效率
- 增强数据分析能力
- 改进用户界面交互体验

3. 架构演进的可能性

- 引入消息队列处理大量日志
- 支持多种数据库存储方案
- 增加实时告警功能

- 支持更多自定义配置
4. 项目过程中的反思与总结
- 技术选型要考虑实际需求和团队能力
 - 架构设计要预留扩展空间
 - 重视文档编写和测试用例
 - 注重用户体验和性能优化

七、其他补充资料（选填）

1. 项目详细文档：[链接到项目 [README.md](#)]