**项目总结报告**

日期：2024/8/9

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 17 | 项目名称 | 任易帮 |
| 编程语言 | Javascript + Java | 开发平台和框架 | React + Spring Boot + Spring Cloud |

|  |
| --- |
| **项目工作小结** |
| 1.采用哪种架构风格？哪些设计模式？  架构风格采用了微服务架构。  设计模式采用了代理模式、责任链模式与命令模式。  2.技术方案有哪些亮点？  (1) 使用Nginx 实现反向代理与负载均衡;  (2) 使用Nacos 实现服务注册、发现和负载均衡;  (3) 使用API网关 集中管理请求路由、安全认证等功能;  (4) 使用sentinel 实现流量监控管理、微服务熔断降级等微服务保护功能;  (5) 使用关系型数据库MySQL与非关系型数据库MongoDB；  (6) 微服务模块实现了多实例部署;  (7) 结合使用华为云的API Gateway和 Function Graph实现了对多实例的崩溃监视与重启;  (8) 使用Prometheus 以及 AlertManager 实现了对后端实例的监视。  3.是否做了单元测试？是否做了系统的功能测试、性能测试、兼容性测试和易用性测试等？  (1) 本项目进行了全面的单元测试，包括前端的Jest测试与后端的Junit测试；  (2) 本项目制定了详细的系统测试用例，并进行了详细的功能测试、性能测试(使用Jmeter)、兼容性测试(测试了在5种不同浏览器上的兼容性)和易用性测试。  4.采用大模型来辅助了开发的哪些阶段（需求/架构/详细设计/编码与调试/测试/计划与报告）？对开发生产率约有百分之几的提升？  采用ChatGPT等大模型辅助开发了需求、架构、编码与调式、测试阶段，对开发生产率约有16%的提升。 |
| **项目组成员对项目的贡献度（%）** |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | 姓名 | 需求 | 设计 | 编码 | 测试 | 项目管理\* | 小计 | | 徐培公 | % | % | % | % | % | % | | 于明睿 | % | % | % | % | % | % | | 王熠笑 | % | % | % | % | % | % | | 吴坤臻 | % | % | % | % | % | % | | 周泓宇 | % | % | % | % | % | % | |

|  |  |
| --- | --- |
| **软件规模** | |
| 前端的代码行数（不包括注解行、空行和复用代码）： | 3427 |
| 后端的代码行数（不包括注解行、空行和复用代码）： | 20481 |

|  |
| --- |
| **经验、教训和建议** |
|  |

项目组各成员签字：