**项目总结报告**

日期：2024/08/07

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 2 | 项目名称 | TimeGenie-人工智能日程助手 |
| 编程语言 | Typescript，Javascript，Java，Python | 开发平台和框架 | Docker，IntelliJ IDEA，Android Studio，VSCode；React Native，Spring Boot，Spring Cloud，Supabase，FastAPI |

|  |
| --- |
| **项目工作小结** |
| 1.采用哪种架构风格？哪些设计模式？  采用微服务的架构风格。运用GOF设计模式对进行详细设计。Facade模式封装方法，Bridge模式接口与实现分离。  2.技术方案有哪些亮点？  使用云数据库实现用户数据的同步。  前端使用asyncstorage库进行本地数据存储，实现用户离线使用。  后端使用springcloud框架+nacos服务注册中心，搭建微服务架构。  后端使用springcache框架，配置redis作为缓存。  后端使用springsession框架，配置redis作为session存储，实现不同实例间的session共享。  使用openfeign库进行服务间调用，同时利用openfeign自带的ribbon实现自定义的负载均衡层。  数据库层面使用mysql进行分布式部署，配置主从复制，实现宕机容灾、数据备份以及负载均衡。  调用大模型作为智能助手，利用RAG实现检索智能生成吃饭推荐。  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？  对前后端都进行了单元测试。系统测试层面，进行了功能测试、性能测试、易用性测试、可靠性测试、安全性测试、兼容性测试、界面测试、压力测试、部署测试。  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？  大模型在各个阶段均起到了辅助作用。在确定需求与架构设计的过程中给予建议；可以给出比编码的基本框架；对于代码错误也可以提供帮助；对于基础的测试可以提供指导。总体而言，估计对于开发有15%的提升。 |
| **项目组成员对项目的贡献度（%）** |
| 合计100%  赵楷越：20%  朱涵：20%  孙恬然：20%  骆镒妤：20%  齐佳怡：20% |

|  |  |
| --- | --- |
| **软件规模** | |
| 前端的代码行数（不包括注解行、空行和复用代码）： | 6404 |
| 后端的代码行数（不包括注解行、空行和复用代码）： | 4511 |

|  |
| --- |
| **经验、教训和建议** |
| **完备的文档和协作工具：**在项目启动阶段就制定并更新良好的技术文档，包括任务分配、系统架构、设计文档、API文档等。借助华为云的任务看板工具，可以清晰地显示人员、任务以及截止时间，大大提升效率。  **团队协作与沟通：**及时沟通项目进展情况，避免进度落后导致巨大差距。除了完成个人任务外，也应关注任务的合理性，及时指出不合理之处，例如可以推迟实现的过于复杂功能。面对争议性问题应当与团队成员深入讨论，确保在决策前充分了解问题，以避免大规模代码重构。保持开放心态，坦诚沟通，鼓励团队成员分享想法和解决方案，促进团队协作的顺利进行。  **技术栈的学习：**学习技术栈需要持续不断地学习和实践，不断尝试新的技术，新技术常常能够带来全新的视角和解决方案，为项目注入创新力量，让其更加完善和具有竞争力。团队成员学习了自己负责的部分相关的技术栈后，应在仓库更新上传学习笔记作为知识共享，也方便后续合作的开展。 |

项目组各成员签字：赵楷越 朱涵 孙恬然 骆镒妤 齐佳怡