**迭代评估报告（简化版）**

　　　　　　　　　　　　　　　　　　评估日期：2024/8/10

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 17 | 项目名称 | 任意帮 |
| 迭代名称 | 完结交付迭代 | 实际起止日期 | 2024/7/20-2024/8/10 |
| 任务达成情况：（完成的任务、实现的功能、进度、质量等）  在本次迭代中，我们取得了显著的进展，完成了以下任务和功能：   1. 完成了所有的性能测试和压力测试，确保系统在高负载下能够稳定运行，并验证了系统在极端条件下的可靠性和响应能力。 2. 完成了所有前端功能的测试，确保用户界面和用户体验符合预期。 3. 学习并成功引入了Nginx反向代理技术，提升了系统的负载均衡能力和访问速度。 4. 通过Sentinel技术对多实例的部署数量进行了优化，确保系统在不同负载下都能高效运行。 5. 通过引入多实例部署，提升了系统的可扩展性和容错能力，并利用Sentinel技术动态调整实例数量，优化资源利用，提升系统整体性能。   本次迭代的顺利完成标志着我们在项目稳定性和性能优化方面迈出了重要一步，确保系统在生产环境中的高效稳定运行。同时，我们持续关注系统的优化和改进，确保最终的产品达到最佳质量。 | | | |
| 问题、变更和返工：（遇到的问题、发生的变更、是否需要返工等）  在这次迭代中，我们遇到了性能瓶颈的问题。通过更改架构和添加多实例的方式，我们成功缓解了该问题，确保了系统的稳定性和高效运行，已经无需返工。同时，我们系统地学习了前端测试技术，克服了学习瓶颈，确保前端功能测试顺利进行并达到预期效果。 | | | |
| 经验和教训：  **性能优化的重要性：**我们在本次迭代中遇到了性能瓶颈的问题，通过更改架构和添加多实例的方式成功缓解了这一问题。这提醒我们在未来的项目中，应提前规划并设计系统架构，以便更好地应对高负载和复杂场景，避免出现类似的性能瓶颈。  **灵活运用新技术**： 我们成功引入了Nginx反向代理和Sentinel技术，提高了系统的负载均衡能力和资源利用率。这说明在项目开发过程中，及时学习和引入新技术能够有效提升系统性能和稳定性。在未来的项目中，我们应保持对新技术的关注和学习，灵活运用以提升项目质量。  **团队协作和沟通**： 本次迭代的成功离不开团队成员之间的紧密协作和有效沟通。在面对问题和挑战时，团队成员能够迅速调整策略并共同解决问题，体现了良好的团队合作精神。在未来的工作中，我们应继续保持这种协作和沟通，确保项目的顺利推进。  **迭代计划和管理**： 本次迭代内容较多，时间紧迫，但通过合理的迭代计划和有效的管理，我们顺利完成了所有任务。这说明在项目管理中，科学的计划和高效的管理是确保项目成功的重要因素。在未来的项目中，我们应继续优化迭代计划和管理方式，提高工作效率和项目质量。 | | | |