**项目总结报告**

日期：

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 14 | 项目名称 | My-Meeting |
| 编程语言 | Javascript为主 | 开发平台和框架 | WebStorm / Mediasoup / Vue / React |

|  |  |
| --- | --- |
| **软件需求特性** | |
| 1.是否实现了项目立项时的所有需求？列出实现的新增需求和未实现的需求。  新增需求：智能字幕和会议纪要（已完成）  未实现需求：美颜  2.采用哪种架构风格？哪些设计模式？  CS / BS架构  事件模式、工厂模式、观察者模式  3.技术方案有哪些亮点？  良好的跨平台能力，提供原生应用体验；选用性能强大、功能专一的框架，精简项目体积。  4.是否做了单元测试？是否做了系统功能测试？是否做了性能测试？是否做了兼容性等其他非功能测试？  是；是；是；是 | |
| **项目组成员对项目的贡献度（%）** | |
| 注：总计100%  蒋哲： 25% 周嘉豪：25%  沈玮杭：25% 陈乐：25% | |
| **软件度量** | |
| 软件代码行数（不包括注解行、空行和复用代码）： | 15335 |
| 复用他人代码行数： | 660 |
| 类的个数： | 57 |

|  |
| --- |
| **经验、教训和建议** |
| 1. 要选择合适的技术栈。选择框架时不仅要考虑效率以及入手难度，还要考虑其项目和开发者社区的成熟度。 2. Git仓库的管理应该严格服从规范，多交流多沟通，从而避免代码混乱。 3. 分工要明确，架构要严谨，避免耦合度过高导致项目返工。 4. 测试很难很重要，尽早开干。 5. 遇到技术盲区或者瓶颈的时候不要紧张，从多角度多方面来解决它。 6. 项目选题应该要更加谨慎，以免选到难度过大的题目而导致开发难度剧增。 7. 可以尝试扩展chromium内核，使其更加适合实时视频流传输。 |