一、主要工作

1、STDO相应模块开发：完成检查报告书和TMF文书管理的开发。 2、列表可拖拽列宽变更：完成列表画面的可调整列宽功能，并优化相应CSS细节。 3、DB生成工具的改修：根据配置文件格式，对DB生成工具进行改修。 4、端末Web管理用画面的开发：端末相应管理画面开发，包括用户管理、设备管理、事务局管理等。

二、收获与心得 1、技术学习：主要学习了Java编程语言和Vue框架，包括基本概念、语法及核心特性。通过系统化的学习和实践，加深对语言的理解，并在后续实际项目开发中不断积累经验，提升巩固自身的技术水平。

2、设计理念：理解了前后端分离的核心思想，即前端和后端各自独立开发、部署，通过API进行数据交互。提高了开发效率，增强了系统的可维护性和扩展性。

3、团队协作：在工作中，团队协作至关重要，能有效推进工作进展并提升个人效率和沟通能力。团队合作不仅是个人成长的途径，更是项目成功的关键。通过相互支持和共同努力，我们可以更快地找到解决方案，避免孤立工作带来的低效和错误，从而提高整体工作效率。

三、问题反思及施策 1、自查不足：对于成果物，自查不充分，存在简单易摘出问题，或存在其他错误和隐患。改进措施：在进行文档和程序类成果物的自查时，需提前做好规划，留出足够的时间。应确保文档中无错别字、语句通顺，业务内容完整且正确，避免遗漏；代码中无编译和书写错误，并通过测试运行保证程序无方向性错误和异常抛出。发现的问题应及时记录并修正，无法完善的做好标记以便后续跟进。做到自查角度全面充分，保证成果物品质。

2、沟通能力提高：在日常工作中，我意识到自己在沟通的主动性和汇报的积极性方面存在不足，同时在倾听后缺乏及时反馈，影响了信息传递的效率和准确性。改进措施：首先，认识到及时主动汇报是工作职责的一部分，这不仅有助于团队协作，也体现了个人的责任感；其次，增强汇报主动性，定期分享工作进展、遇到的问题及解决方案，并评估是否需要相应调整工作计划；同时，提升倾听与反馈质量，确保即时确认理解，并给予清晰、具体的反馈，提高沟通效率。最后，定期总结沟通中的问题，反思经验教训，提升细节处理能力，避免类似问题再次发生。

3、设计不充分：在设计初期，由于对需求理解不充分或考虑不周全，导致部分功能被遗漏、数据结构选择不当、接口设计不合理等问题。这在后续开发过程中需要额外的时间和精力进行补充和完善。

改进措施：在纸芝居和基本设计作成后，通过画面和设计书，要在脑海里模拟一下程序运行，重点关注需要什么数据，明确哪些数据需要从数据库或外部系统获取，是否需要调用相应API接口。据此完成详细设计，然后可以找老师或者同事帮忙检查一下，确保最终的设计方案既满足功能需求，又具备良好的扩展性和维护性。

4、不注重编码细节：在编码过程中，有时不注重细节，特别是在处理数据类型、数据结构和字段名称时容易出现疏忽。例如，字段名称的不一致性（如使用 studyId 还是 id），可能导致数据传递错误或难以维 护。这种不一致不仅影响代码的可读性和健壮性，还可能引发潜在的bug。改进措施：代码自查时，确保选择的数据类型和数据结构适合具体的业务场景，避免不必要的复杂性和性能问题。如果由于其他限制无法统一命名，在代码中应明确转换为一致的名称，使代码含义更加清晰。