在这次的IDE开发过程中，我主要负责整体的技术把关，解决开发中的各种技术问题，也具体实现了查找替换、代码补全、注释隐藏显示、程序调试等功能。这其中最让我自豪的无疑是自己从最开始“这是啥，怎么搞”的一头雾水，到后面逐渐掌握，实现越来越多的需求。如今项目开发告一段落，我也从实践中体会学习了许多，在此记录一下：

**1.团队配合**

这次开发过程中，我们是以5人团队的形式来开展工作，我们在一开始其实没有完成明确的分工，因为大家熟悉的领域有所不同，而这次要做的项目的综合性比较强，所以还是先给了大家自己开始学习并逐步实践的时间，给大家的自由时间能够让大家更充分地发挥出学习能力和编程能力。

此外，我们在任务后期的攻坚阶段大量使用结对编程的模式，这虽然看似降低了整体的效率，浪费了人员，但这明显提高了代码质量，让我们的代码稳定性提高了很多，这在时间紧张而人员不用太分散的时候非常有用，保证了任务的稳定高效。

**2.问题交流**

尽管开发过程整体比较顺利，但还是出现了小插曲。在我编写调试功能的时候，需要使用QProcess调用gdb进行调试，然而QProcess与gdb的交互却始终存在问题，耽误了不少时间。好在最终问题成功解决，算是虚惊一场。在此有必要总结一下经验。

在当初遇到问题的时候，我尝试过阅读帮助文档、百度搜索错误提示等方法，然而这些并不奏效。在万般无奈的情况下，我们向老师求助。尽管老师也并不擅长Qt开发，没能直接帮我解决问题，但他为我提供了一个很重要的思路——到技术论坛提问。于是我来到Stack Overflow上贴自己的代码提问，得到了大家的回应。也正是在和全世界的程序员的交流下，判断出应该与文件输入输出的缓冲区有关，最终解决了这个难题。

总结一下，尽管程序员是我们心目中典型的理科生，但并不代表文科的技能就可以被忽视。资料查找和信息交流便是一种非常重要的技能。虽然错误提示是类似的，但产生的原因却不尽相同，我们不能单纯靠照抄别人的代码解决问题，而应该积极和别人交流，在思维的碰撞中找到灵感。

**3.顶层设计**

这次的项目开发还是存在着一些不足，最突出的问题就是欠缺顶层设计。在我们查阅海量资料后，发现了Qsciscintilla这个工具库。在这个方便的工具库的帮助下，我们迅速搭建好了项目框架，在很短的时间里就实现的大部分的基本功能。但也导致软件系统虽然功能比较完善，但是其代码质量和整体架构是不足的。最突出的影响就是导致我们在后续添加标签页时举步维艰，经过估算其工作量与从头开始无异，这个功能最终也只能无奈割舍。这启示我们，在将来的项目中，一定要在开始做好顶层设计，将系统的架构搭建起来，这样才能更好地将系统分模块，便于后续结构、功能上的维护。

**4.规范文档**

必须要承认，写文档是一个痛苦的过程。不过好在我们在最初着手编码前便约定好代码注释要求，开发过程中产生的临时文档也被一一记录，很大程度上减轻了最后编写文档的压力。经过这次的项目，我们都认识到了代码注释的重要性，决定在今后时间更充裕的项目里严格执行代码注释约定。这对未来的开发起到很大帮助。

回想起这个小学期，我们曾经通宵编程改代码，也提前完成各项需求，提前开启“假期”生活。这其中，要感谢自己的辛苦付出，感谢老师的耐心指导，更要感谢队友们的相互支持，通力合作。总的来说，我在这个小学期学到了很多，希望这些能对将来学习和工作有所帮助。