

fileEditor 中期报告

我们的选题是「文本编辑器」，我们暂时称之为 fileEditor。在中期报告中，我们主要介绍当前完成的讨论、预期、调研和设计。

基本设计

开发环境 我们使用 Qt6 作为图形化界面的框架，使用 CLion 作为 IDE，使用 GitHub 管理仓库。

界面设计 我们尝试模仿 Typora 的较为简洁的界面，它只包含标题栏、菜单栏和编辑区域。对于任何一个 fileEditor 工作进程，它负责且仅负责一个文件的编辑。因此，fileEditor 的前端工作事实上比较少，但一个重大难点是编辑区域的实现。

文字格式 我们计划在 Markdown 标记语言的基础上支持文字格式设置。我们参考了语雀和 Typora 的「所见即所得」的编辑理念，计划支持按键、快捷键和 Markdown 语法三种设置文字格式的方案；主要支持标题、加粗、斜体、颜色等格式。对正文文字大小和字体的支持在实际场景中应用不是特别广泛，因此我们可能不支持这些功能，或者仅在段落级别上支持这些功能。

排版设置 我们计划支持对齐方式的设置。关于题目中「自动分页」的要求，我们预期的实现方案是，使用类似 Microsoft Word 的分页方式，即固定页面的长宽，而允许用户调整页面的放缩比例。我们也预期支持用户自定义页面的长宽，但如果实现比较困难的话也可能并不支持。

实现细节讨论

文件保存方式

我们有两种文件保存的实现思路。第一种是将对象序列化后存入文件中，第二种是以类似 Markdown 的方式存入文件中。后者的优势在于文件可以使用其他方式打开，但劣势在于处理标记时会比较复杂，并且涉及到转义等问题。前者则在实现上可能比较简单。我们将在后续具体实现过程中完成实际的选择。

文件的显示

经过调研，我们初步计划使用 `QTextEdit` 实现富文本内容的显示和编辑。文件后端计划采用 `QTextDocument` 实现。`QTextDocument` 事实上支持了大多数需要的功能。如果确有需要，我们可以考虑自行补充一些功能的实现。