

1) 参与访谈人员

访谈者: 刘瑞芳,刘馨蓬

受访者: 谢宇轩

2) 访谈信息

- **最近使用...ide在做什么? 使用时更注重什么功能**

主要用来写小程序, 编写较大的项目, 编辑、编译、调试代码, 导入写好的代码

- **在使用过程中有什么较好的使用体验, 或有什么满意的地方**

Dev c++ 使用过程中较为轻便, 打开快速, 编辑测试小段代码方便

vs 在项目管理上比较好, 可以很好编辑代码, 会补全代码, 自动对齐, 可以检索库函数参数表, 源码等, 支持多种语言, 提供了大量的插件, 扩展性强, 可以定制ide功能, 自动生成类的声明和定义。

- **那您使用遇到过什么问题, 令您不满意的地方有哪些**

vs 多文件编辑较为繁琐, 很多功能例如文件切换编辑依赖鼠标, 编写效率下降, 报错复杂, 没用的信息多, 启动慢, 代码优化不合理, 选项少且设置复杂容易出现意料之外的bug

- **您觉得ide更为智能的特点应体现在哪**

1. 代码补全、自动对齐
2. 自动生成框架
3. 去掉不需要加载的模块
4. 错误报告时记录当前的状态信息
5. 代码托管, 可以与代码托管平台等建立更为方便的对接
6. 自动保存
7. 效率更高的优化

3)需求

- **分析**

客户描述	需求	需求类型
编写小程序，大项目	开发小程序和软件	业务需求
打开快速/启动慢	效率高	非功能性需求
项目管理	文件管理功能	功能性需求
编辑代码，自动补全，对齐，检索功能	文本编辑及辅助	功能性需求
支持多语言	可以用来编写各种语言的程序	功能性需求
提供插件	可扩展性要强，提供多样化功能	非功能性需求
自动生成类的声明和定义	能够根据信息创建类和简单定义	功能性需求
轻便/文件切换繁琐，依赖鼠标	多设置简单快捷键操作	功能性需求
报错复杂	优化报错分析，提供简单和详细两种	功能性需求
编译，优化，选项设置少	用指令和多选项进行编译	功能性需求
导入代码	识别并导入新的源文件	功能性需求
调试代码	有调试的功能	功能性需求
自动生成框架	框架构建	功能性需求
去掉不必要模块	智能编译	非功能性需求
错误报告储存状态信息	出错状态保存	功能性需求
代码托管	云保存代码	功能性需求
自动保存	编辑时自动保存	功能性需求
效率高的优化	编译优化	功能性需求

• 整理

1. 业务需求

开发小程序: 便捷编写小程序,测试代码等量少的代码

开发软件: 方便管理项目

2. 功能性需求

文件管理、文本编辑、**辅助编辑: 自动补全,代码对齐,代码检索,字符串替换等文本编辑功能**,更方便的编辑和管理代码

报错信息简单易懂、多语言识别和编译、**调试功能、编译功能**: 增加选项进行编译优化, **程序运行出错时自动保存出错时的状态(寄存器等)**, **智能链接**, 将引入的模块仅链接入使用的部分代码, **减少程序大小。代码优化时间和优化质量尽量优化**

根据信息创建类和简单定义、**自动生成框架**: 添加类的时候自动生成对应的头文件和源文件, **包含简单的声明, 对于一些基本框架(类似mfc, qt等)可以自动生成**

对常用操作设置更多简单快捷键、

识别导入项目外文件、**文件管理, 代码托管**: 拥有自己的代码托管平台, **远距离多人开发方便本地更新和上传**, 对于项目文件有良好的管理方式(好的视图等), 可以引入已经成型的文件和项目

3. 非功能性需求

效率高：软件启动速度快

可扩展性强：拥有多种功能插件，支持功能扩充