

软件需求工程

Homework 2

161220056 敬舒舒

2019 年 10 月 17 日

1 访谈参与人员

需求分析员：敬舒舒 张梦窈

项目负责人及经理：吴雨昕 李培凯

2 访谈信息

Q：请问你们的公司名称及各自职务是？

A：上海飞讯科技有限公司。我们是项目负责人和经理。

Q：请问你们想开发一个怎样的软件？

A：我们想开发一个类似于市面上已有的、比如类似于 vscode 的 smart IDE，它的体量不是一个非常重量级的 IDE，不像 visual studio。它至少具有健全的文本编辑功能、自动补全功能，可以支持大量扩展插件，支持市面上主流的高级语言，如果可以，最好能够支持汇编之类的相对底层的语言。我们做这个 smart IDE 是希望能与市面上的一些已有的 IDE 做出一些创新型的区别。

Q：请问这个 smart IDE 主要支持小型还是大型工程？

A：这个 smart IDE 可以非常小，但是它对于一些需要扩展性的行为可以通过支持插件的方式将它扩展得功能非常强大。

Q：请问你们公司的业务是？

A：我们主要是做嵌入式开发，这也是我们希望这个 IDE 能够支持汇编等底层语言的原因。

Q: 请问您是希望这个软件像 `vs code` 那样支持 `windows`、`linux` 等平台吗?

A: 在版本初期只需要支持主流操作系统就可以, 比如 `windows` 和 `mac`。`linux` 暂时不需要。

Q: 之前提到了希望支持插件, 那请问插件是由我们来做还是您另外找其他公司或者是贵公司自己做?

A: 我希望你们能够提供一些基础的, 保证 IDE 正常运行的插件。至于其他的插件可以交由爱好者或者其他公司制作。

Q: 请问这个 IDE 做好后您是打算开源吗?

A: 这个出于自主知识产权的目的, 我们暂时打算自己使用。因为这个 IDE 使用范围只是针对我们公司的, 其他人未必用有这样的专业性需求。

Q: 关于开发周期您有什么要求吗?

A: 我希望半年至少出一个可用的版本, 之后在一年时间内多次迭代, 最后达到一个满意的效果。

Q: 这个项目是打算长期维护吗?

A: 我们打算长期维护。

Q: 这个 IDE 是打算公司内部使用, 是不是需要加上一些协同开发的功能? 充分利用内网及内部服务器。

A: 这个应该是通过扩展插件就可以实现的, 应该属于基础插件的部分。另外对于我们嵌入式开发, 希望在调试的时候能够把调试结果可视化。

Q: 可视化是指把参数变量都打印出来, 还是希望把运行情况展示出来

A: 我们希望能够监视它运行时候的内存占用、有没有内存泄露的情况、包括运行时刻的突发异常, 我们希望这些都能够实时显示出来。

Q: 那么您是希望这个 IDE 支持市面上所有的主流语言?

A: 我们希望优先支持适合嵌入式开发的语言, 比如说汇编语言、C/C++。

Q: 请问您对这个 IDE 的外观有什么要求? 比如 `sublime` 和 `vs code` 更喜欢哪种风格。

A: 我认为 `sublime` 不如 `vs code` 好看, 偏好科技感的外观。

Q: 那么我们会把基础外观作为选项放入工具栏, 其他特殊效果或者“皮肤”则使用插件。

Q: 请问您能接受哪种付款方式?

A: 我们打算最开始支付一部分定金, 第一版本上线可用再支付剩余定金。持续迭代阶段再每月支付部分报酬, 后续维护费用另算。

Q: 请问您对开发方式或流程有什么要求吗?

A: 前半年在第一版本上线之前我们不干涉开发, 只要半年的时间提供给一个可用的版本。持续迭代的时候只要设定一个迭代周期持续汇报就行。

Q: 请问您还有什么补充的吗?

A: 可视的文件结构和简洁美观的工具栏。

3 需求整理

3.1 业务需求

上海讯飞科技有限公司希望开发一个针对嵌入式开发的、内部使用的 smart IDE。

3.2 用户需求

- a. 上海讯飞科技有限公司的工作人员能使用该 smart IDE 进行嵌入式开发。
- b. 上海讯飞科技有限公司的工作人员能使用该 smart IDE 进行协同开发。

3.3 功能需求

- a. 文本编辑
- b. 自动补全
- c. 支持主流的高级语言
- d. 支持相对底层的语言 (如汇编语言)
- e. 支持嵌入式开发
- f. 支持扩展插件
- g. 提供基础插件 (外观调整, 协同开发, 调试结果可视化, 监视运行状态)

补充的功能需求

- a. 高级的自动补全功能：根据上下文补全变量名和函数名，按推荐度排序，而非按字母顺序排序。
- b. 框架代码推荐：联网状态下，根据输入搜索并推荐对应的框架代码。
- c. 编写汇编代码可以自动适应平台，选择对应的指令集。
- d. 自动配置编译器：根据当前计算机的环境，自动设置并优化编译选项。
- e. 可视化的类关系图：体现各类之间的依赖、继承等关系。
- f. 代码拼写检查：提示可能的拼写错误，允许用户忽略该错误，同时也允许用户关闭该功能。

3.4 非功能需求

- a. 科技感外观 (可参考 vs code)
- b. 可视文件结构
- c. 简洁美观的工具栏

3.5 设计约束

- a. 支持 windows 和 mac 操作系统
- b. 非大体量

3.6 商业约束

- a. 半年交付可用的第一个版本，接下来一年内迭代开发，最后需进行长期维护
- b. 报酬分四个阶段支付：最开始支付部分定金，交付第一个版本后支付剩余定金，持续迭代阶段每月支付部分报酬，后续维护费用另算。