

# 实验1 需求获取

---

## 1.小组成员

---

- 宋超群 —— 171860642
- 童逸舟 —— 181860090
- 郑浩然 —— 181860147
- 李舟俊潇 —— 181860051

## 2.小组成员任务及成绩分配

---

### 宋超群 — 25%:

使用爬虫获取数据，并根据词云分析的关键词筛选最终的数据

### 李舟俊潇 — 25%:

`VSCodeQuestions` 部分词云分析后进行需求分析并整理分类

### 童逸舟 — 25%:

`VSCodeIssuesTitle` 部分词云分析后进行需求分析并整理分类

### 郑浩然 — 25%:

`PullRequest` 部分需求词云分析后进行需求分析并整理分类，报告撰写

## 3.实验目的

---

`vscode` 是一个知名的开源项目，在这个项目中有大量的 `Pull Request`，同时 `Stack Overflow` 上也有很多与其相关的问题以及这些问题的优秀解答。为了更加深入的了解 `vscode`，我们选择通过爬取 `Github` 和 `Stack Overflow` 上与其相关的内容并对其进行了词云分析后提取与高频词汇相关的需求进行了分析和分类。

## 4.实验方法

---

首先通过自己设计的爬虫程序对 `Github` 和 `Stack Overflow` 上对关键词 `vscode` 的所有 `Pull Request` 和 `Issues and Question` 进行了爬取获得了 `PullRequestTitles`，`VSCodeIssuesTitle`，`VSCodeQuestions` 三个文档。由于数据量非常大，有的文档甚至多达万行，因此必须对其进行词云分析，找出其中出现频率最高的词汇后二次整理文档，筛选出覆盖高频词汇的那些需求再进行分析。由于这些需求都为英文，因此我们首先需要将需求翻译为中文，在对其进行需求分类，最终汇总出了三个分类文档。

## 5.实验结果及效果分析

首先我们对三个初始爬取文档进行了词云分析：

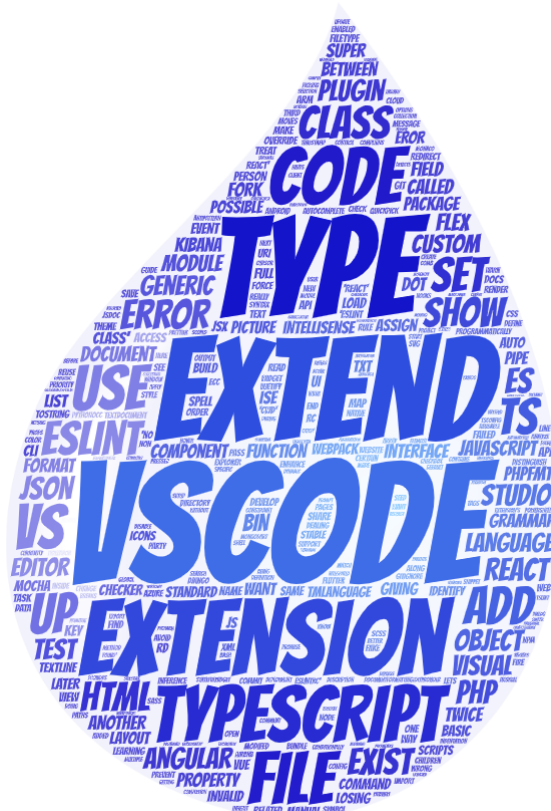
PullRequestTitles

Filter	Size	Color	Angle	Font	
Add	35	Default	Default	Default	▼
Extension	16	Default	Default	Default	▼
Support	16	Default	Default	Default	▼
File	12	Default	Default	Default	▼
Git	11	Default	Default	Default	▼
Added	10	Default	Default	Default	▼
Allow	10	Default	Default	Default	▼
Editor	9	Default	Default	Default	▼
Show	8	Default	Default	Default	▼
New	7	Default	Default	Default	▼
Variable	7	Default	Default	Default	▼
Command	7	Default	Default	Default	▼



VSCodeIssuesTitle





经过词云分析之后，我们选取每个文档中的高频词汇对文档进行筛选，只保留含有高频词汇的需求，经过翻译后分类最终得到了三个文档 `PullRequest中文分类.docx`，`vscodeIssues中文分类.docx`，`vscodeQuestions中文分类.docx`，这些文档可以在我们的项目目录中进行查看。

我们将这些需求分为了三类，分别为：代码编辑（编辑界面优化、关键词联想）；代码 & 文件管理（Git、查找引用、文件存储）；调试 & 运行相关。经过我们对三个文档的统计，涉及三种需求的条目个数分别为  $46 + 22 + 16 = 84$ ， $35 + 13 + 19 = 67$ ， $12 + 16 + 8 = 36$  条。这些数字可以直观的表现出大家关于 `vscode` 的需求，即代码编辑界面上，可优化的地方更多，通过众多的 `Pull Request` 可以使得程序员在写代码时获得更好的编辑体验。而代码和文件管理方面同样还有很多可以改进的地方，因为Git总是不可避免的出现很多难以预料的问题，着实困扰了很多非专业人士，通过这些改进也使得 `vscode` 更加的亲民。关于调试 & 运行方面的需求较少，据我们的分析，一方面的确是因为调试方面 `vscode` 已经足够优秀，并且调试方面的问题相对较为单一，因此很难提出改进。另一方面，调试所涉及的知识也更加专业，因此相较于代码编辑，文件管理拥有更高的上限，一般的程序员很难就此提出改进。但是这也恰恰说明这些少有的 `Pull Request` 的重要性，一般都关系着软件的核心。

## 结论

通过本次实验，我们小组更加深刻的认识到了如何获取软件的相关需求。通过对众多需求的深入分析，我们意识到即使是如此完美的 `vscode`，也有那么多的需求可以改进，也让我们侧面了解到了需求的重要性。正是因为如此多的需求，才使得 `vscode` 在竞争对手众多的市场中不仅没被打败，反而更加优秀，成为被广大用户推崇的编辑器。