**P4**

2021-10-3 10:0:34

【问】

好的，那我们现在开始吧，嗯，首先就是想了解一下你的基本信息，你能说说你的专业背景和做开发的经历吗？大概是有过几年的开发经验？

【答】

哦，我现在是21级XX专业的学生，然后我本科也是计算机方向的，就是，然后大三就是20年是第1段实习是在XXXX，做的是后台开发，这边从事主要是XX音乐里面的直播的一些后台模块的开发，然后是今年就是21年4月份也是在XX，也是后台开发，这边主要就是XX底层的数据库系统的一些组件的开发，对主要是这样的。

【问】

那相当于你是有了大概三年的开发经验了？

【答】

算是吧，但其实因为我是计算机专业的，在学校有那些也做过一些项目，就是课程作业啊，项目之类的这些，那种也可以算，好那可能是算四五年。

【问】

那就是我们这个研究，我们是针对的是中国的开发者，我们是想了解一下就是中国的开发者的一些特征，所以呢，就想问一下，你觉得在中国程序员这个群体和其他的专业，其他职业的人相比有哪些特征？有哪些不一样的地方？

【答】

这个问题啊，程序员，首先是算是现在比较火的一个职业吧，所以他们的工资可能相对高一点。哦，但是加班也很严重，所以时薪可能并不算很高。还有一个就是，就程序员其实也是按照，就是比如说产品经理或者什么布置下来的需求来去做一个概念的实现这样一个工作。

【问】

比如说从性格啊，还有思维方式，思考的方式以及一些学习的方式，学习的习惯上来讲有什么不同？

【答】

因为程序员大部分时间就每天工作都是面对着电脑，可能他们的，当然不是所有啊，就是他们可能会比较偏向内向或者是有一些那种理工男的那种比较直率的思维，就不是有个段子说是什么，就是买西瓜的那个段子你应该听过，就是让程序员去买菜，如果看到卖苹果的时候买两个，如果看到卖西瓜的就买两个，结果那个人回来就只买了两个西瓜，原因是因为他看到了苹果，就差不多是这种就是比较像是程序一样，一条线有固定的逻辑，这样走下来的一种思维。

【问】

嗯，然后学习呢？学习方式，就是平时有自己独特的学习的习惯吗？

【答】

因为开发语言还有技术这一块发展的比较快，所以作为程序员需要经常性的去学习新知识，所以一般程序员的学习能力可能会稍微就是都要比较强一点，不然可能会在技术这方面就会容易遭到淘汰啊什么的。

【问】

嗯，那你们学习是倾向于自学吗？

【答】

嗯，对，包括包括在学校的时候，可能大部分的因为老师教的东西可能更多是概念性和系统性，或者是比较老的一些东西，所以要想就是从一个软件方向的学生变成一个程序员的话还是需要大量的自学的。

【问】

好明白了，嗯，然后就是简单的讲一下你的一些开发经历吧，哦，首先是请你回忆一下最近学了什么，也不一定是最近吧，就说你学过什么新技术新的工具或者软件，就举一些例子，然后讲一下大概的讲一下你的整个学习过程是怎么样的，就是说你从听说这个这个软件开始到最终学会了这个软件这个工具或者是这个框架，然后学会它之后用它来进行开发，完成你的一些开发的工作项目，整个过程是怎么样的？

【答】

嗯。就拿最近一次实习，在就是XX那边，他们是做数据库存储的，然后相当于是我进去以后就是导师他们让我先去看一些，一个是谷歌的一个开源的数据库的库，叫LevelDB，它是开源在GitHub上面的，就是因为XX，就是在实习那边，他们底层整一套是基于这个东西来做的，所以我作为一个新入职的实习生，就需要先去学这个东西。然后学的方法就第1个是看那个他的文档了，它有那些quick start，还有一些详细的说明，然后导师就让我照着他们就写几个小demo这样子，可以相当于快速的认识这个软件是做什么的，有哪些接口，然后，嗯，这是怎么用的，然后写几个demo以后，就只看，就是自己写的只是一些非常小的demo，然后下一步应该就是看老师给我的，就是他们正式员工以前写好的一些代码，就是这个组件在他们的产品中是怎么用的，相当于一开始还是看和试这样的操作，然后才慢慢的自己接一些小需求，然后再逐渐的用这个组件的越来越深入的功能吧，然后去，然后更加熟悉这样。

【问】

嗯，那你就是在这个学习过程中，其实就是用到了这个相关的资料，是吧？文档，嗯，那就是你能具体的说一下都用到哪些文档？

【答】

文档主要就是，因为它的软件是开源的，它本身就带有文档，就是官方文档是一个，但是，官方文档因为是英文的，然后我们就是实习要尽快的上手嘛，所以还是会搜一些中文的资料，一个是文档对应的中文翻译，还有一些是中文博客，就是不是官方的那种文档，就是第三方的，但是写的也比较系统的这种第三方文档吧算是，就是这样子。一个是官方文档，一个是第三方的比较系统的文档，还有就是一些博客资料啊这种。

【问】

这种第三方的文档，它是由专门的机构来写吗？还是说个人感兴趣，然后就把它翻译出来写的。

【答】

那种相对系统一点呢，我觉得应该也是有，也算是啊这种开源爱好者吧，就是自发组织起来的，应该不是个人，但是参与的人可能也不是很多这种。

【问】

但还是有个团队来专门做的，那你认为文档对于开发工作的重要吗？

【答】

这个肯定是非常重要的，因为你遇到一个新的软件，新的模块或者是新的框架，嗯，肯定是需要文档来让你知道这个东西是怎么用。因为遇到一个新东西肯定就是一开始是先会去用它，然后就是，不可能是一开始就去深入去看这个软件是怎么做的，去看他的代码，所以文档肯定是非常重要的。

【问】

就是文档的作用，一是让你开始就在不知道的情况下，让你先大概的了解一下这个软件，对吧？那么再就是你熟悉了之后，在具体的开发工作中，就比如说在开发的中间的阶段你还会用到文档吗？

【答】

这个也是会的，因为我觉得文档两部分，就是两个比较重要的功能，一个是他文档可能会有一个快速上手的环节，让你快速知道，这个比如说软件，有哪些基本的，有哪些基本的模块组成，然后他最基本的用法是怎样的，另外一个就是他可能会，就是文档的主体部分，就后面会有那种非常长的章节介绍的，软件中的每一个组件的详细的功能，然后实现方法，还有接口的这些参数啊什么乱七八糟的，这种就是我们在已经熟悉这个软件后面去开发的时候，因为我们不是这个开源软件的开发者，就是肯定不能，不会对它里面每个细节都很熟悉，在用到某个地方发现这个地方好奇怪，或者是遇到什么bug就会去，有目的性的去看那个文档的某一个章节。

【问】

好，大概的了解了，就是说明你对文档的这个结构也有一定的认识，应该也是看到过比较多的文档吧。

【答】

对我感觉那种质量比较好的文档都会有这种quick start.

【问】

嗯，你能举一下你看过的文档的例子吗？

【答】

哦这个。好像还挺多的，这个举例子一时间，想一下。嗯，比较好的可能还是这种像Google ，Facebook这种国外的大型公司做的开源软件，比如一个是我刚刚说的那个那个MongoDB，还有Facebook的RocksDB，当然这些都是数据库相关的。嗯嗯，但是国内的也有，比如那个XX公众号后台开发也有，他有那个文档，但那个文档感觉就不是很好吧感觉。

【问】

那你能就具体的说一下，你觉得好的这些文档它好在哪里，然后不好的文档他不好在那里吗？

【问】

嗯，好多文档一个就是像我说他会有那个quick start，让你快速上手这个软件，像网上说的XX公众号后台好像就没有这种东西，他就是一上来就是那种就是这个功能是做什么的，它的实现方法那种，又臭又长的，你一开始就很难去看进去的那些东西。还有一个可能就是文档的规范性，比如说他们现在很多文档都是用那个Mark down写的，就是Mark down是一个排版的预言吧。就是作者在写的时候就只关注内容，然后它会自动生成一些排版，这样的好处就是生成出来的文档就非常的规整，就是什么缩进空格还有表格的样式都是非常统一的，就是你看起来就非常舒服。像我刚刚说的XX公众号后台的这个文档就感觉就写的比较随意，就比如同一个表格，他有时候表格是用的截图，就是表格本身是个图片。偶尔里面还会有拼写错误，错别字，这种就感觉不是很规范，就看起来不是很舒服。

【问】

那么就是从内容的全面的程度上来看他们有什么特点，内容全不全？

【答】

嗯，好的文档肯定还是会比较全的，他甚至会每一个细节都会给出一个例子来便于理解，不好的文档，他有可能就是指把那个比如说一个函数的接口参数丢上去，就是告诉你有这些参数，但是不告诉你这些参数是做什么的，就需要，甚至还需要自己去去试一下，就填各种不同的参数，去试一下才知道它是什么功能，反正就是不是很好的文档。

【问】

那从文档这个结构，组织的架构上来看，好的文档有什么特点，不好的文档有什么特点？

【答】

嗯，组织结构好的文档肯定是就是他给文档分的章节还有模块都是非常清晰的，然后可能那个页面旁边还会有一个总的目录大纲，导航栏。不好的文档这方面可能就做的比较随意，它只是简单的功能罗列，可能就没有去细心的去整理这样子。

【问】

说的这个不好的文档还是XX的那个例子吗？还有就是国内还有没有其他例子？

【答】

哦，还有就是企业内部，就是不是开源的那种，面向内部的文档，用文档就非常不好。就相当于可能只有写这文档的人自己才能知道这个文档的结构还有组织什么，其他人看的话就会感觉有些莫名其妙，有时候还要去问情况的那个人。

【问】

这种文档是你在实习的过程中看到的是吗？就只能是在内部看，他没有公开的放到网上？这种文档是pdf的吗？他不是那种网页？

【答】

都有，一个网页，就企业内部他一般会有那种文档的平台之类的，就是会也是写在上面也是纯文本的，，但是由于是旧开发者自己写的，也不是很规范，写的就比较随意，不同开发者的习惯也不同，写起来，不同人写出来的文档的那个组织方式也不一样。

【问】

就刚才也提到一些国内国外的这个文档的例子，那阅读的过程中就会涉及到语言的问题，你觉得语言问题，就你在读的过程中，你觉得语言是一个障碍吗？

【答】

嗯，对于就是纯粹的开发文档，其实还好，没有那种什么文学作品啊那些难读，因为大部分还都是。怎么说呢，就是你看我那实际也不需要去逐字逐句的去看他的，就是一些什么语法问题这些东西，就算不是很懂也不影响理解这个技术方面的东西，所以应该相对来讲还好，但是看英文的可能还是会慢一点，在需要你快速的去，比如说，要快速的去做一个需求或者是什么作业的ddl快到了，这种就开始，可能会去寻找一下中文的，中文看起来还是更快。

【问】

嗯，那下一个问题是你在开发过程中应该是会遇到一些问题，那么遇到问题的时候你会怎么办呢？是会先去自己查看文档还是说问别人？

【答】

第1步应该是先去看文档，如果文档上找不到那个，找不到自己想要的问题，第2步就是去搜索，就是直接去拿这个问题在Google或者什么地方去搜索这个问题，然后最后才是会去问其他人。

【问】

嗯，那就是比如说遇到一个bug，你是倾向于自己回去，就第1步是先看文档，而不是直接搜这个bug，这个错误的代码？

【答】

嗯，这个可能也是会看情况。比如遇到一个bug，然后他就这个bug会给出一个什么提示，这种情况下可能会直接去顺手去把这个提示给复制到Google里面去直接搜这样子。然后如果是在开发的时候。就是到一个地方，突然不知道该用什么，这种一般会去看文档。

【问】

遇到一个比较细节的问题，然后就会看去查看文档，对吧？

【答】

嗯，也不是细节问题吧，就比如你正在写，但是具体的例子可能暂时想不到，就是比如说你正在写一个程序，突然有一个地方，就是你在用别人的框架写一个程序，突然有一个地方你知道你想实现一个什么功能，但是不知道对应到这个框架里面应该用什么样的模块或者什么样的函数，这种会去看文档。这种系统性的问题或者是面对有一些未知的这种问题，前面说这种是会看文档，然后去直接去查，可能是比如说运行的时候它直接出来了一行报错，就是有具体的报错内容，就是这种直接搜这个报错的内容肯定会比较快一点的。

【问】

那你平时都会用到哪些文档？刚才提到了官方的，官方的那种文档，那么除了官方的，还有其他的途径吗？

【答】

一种就是官方文档，然后有些著名的，著名的开源软件，会有很多爱好者维护着不同的文档，就是一种是比如国外软件官方文档是英文的，国内会有一批爱好者维护中文文档，但这个中文文档其实也不是官方的，就是他们自己因为爱好，然后把这个文档翻译了一下。

【问】

那除了这个还有其他的一些吗？比如说你会你再学一个新东西的时候，你也会去看相应的教程，教材之类的吗？

【答】

对，也是会有的，像那些，国内有那个菜鸟教程，上面就是有各种编程语言的教程，然后各种常用的软件框架的教程上面都有，那个也是我算是经常看的，但那个也不是官方文档，它其实感觉和文档还不太一样，它是那种教程，相当于感觉像是一种扩展版的那个quick start，它主要是一些很多demo组成的，还有一些例子，这样子告诉你，一步一步告诉你这些东西怎么用，是一个比较详细的这种教程类的。

【问】

那刚才也提到博客，有的人在分类的时候也其实也把它们分做文档，就技术文档，那你会看博客，博客或者是一些嗯社区吧，就是程序员、爱好者他们自己维护的这种社区里面的问答等等。

【答】

嗯，这种肯定都是会看的，因为这种功能肯定要，这个里面含的内容会比文档，还有教程要丰富得多，而且会有很多都是非常具体的，像刚才说的，遇到每一个报错，你基本上一搜就可以搜到那些问答社区里面，就是有人遇到了相同的问题，在上面提的问题，以及其他人的解决方案，就是这种覆盖面会非常的广，然后就是基本每一个细节的问题都会有相应的，都能查到相应的条目，

【问】

那这种博客跟官方文档来比，你觉得他们的质量怎么样？

【答】

可能总体来讲肯定还是官方文档的质量会高，然后博客的质量会相对低一些，但是像国内有些这种XX后台啊，他文档本身写的就比较差，然后也不详尽，这种的话，可能博客反而会好一点，但博客有个问题是什么，就是可能不同的人遇到相同的问题，造成这个问题的原因是不一样的，如果在看博客的时候，有时候想去按博客一步一步的，就按博客里面一步一步操作下来就会发现自己的结果和博客里面的结果不一样，比如博客成功了，但自己还是照着他一步一步做，还是没有成功，这种可能会是一些由于环境啊什么，就是差异化的问题可能通过博客就不是很容易的去解决吧。

【问】

就比如说官方你遇到问题的时候去看官方文档啊，官方文档没有解决就会去看博客对吗？

【答】

嗯嗯，嗯，博客其实也是通过搜索找到的，官方文档就可能不是搜索，就是会有目的性的去找。

【问】

那提到搜索就是你一般通过什么方式去找，去找到需要的技术文档，除了搜索之外。

【答】

那就是包括博客这些在内的吗？可能还是直接搜问题吧，比如如果有报错就直接搜报错的那句话，或者是自己想查什么就会拿那些具体的关键词去搜索。

【问】

可能是直接进入到那个官方文档的网站，对吧？

【答】

据说从搜索进入官方文档也是会有。但是，你已经知道了这个文档是，已经知道了你想查的是哪个文档，可能为你直接知道网址，甚至可能会直接输网址，或者是就算搜索也只是为了图方便，就是快速的进入到这个官方文档的网页而已，这个可能严格来讲和搜索还是不太一样，就没有利用到它搜索的那个核心的功能吧，这是作为一个快捷入口。

【问】

那找到了这个文档以后在阅读的时候，嗯，应该是这个文档应该是一个体量比较大的一个，就有很多页很多内容的东西，那你是怎么快速的找到自己需要的信息？

【答】

首先一个是导航，一般比较好的软件的导航都是非常清晰的，第1章是快速启动，quick start，然后第2章可能是一些就是概述吧，就是一些术语的解释，还有一些软件，整个软件是总体是由哪几部分大模块组成的，然后后面才是每一章可能是一个小模块或者是一个小功能，这样然后所以通过导航可能会比较方便的找到自己想要的内容，如果没有的话，就是会去搜索，因为搜索引擎一般会有一个，就是那个可以输入你想查的内容，后面跟site冒号加一个网址，相当于就是在这个网站里面搜索，就是在文档里，如果你很难自己找到，就是自己不能直接通过导航找到自己想要内容的话，就会通过这种方式，在文档的网站内部进行搜索。

【问】

文档的网站内部，他有可能也有自己搜索，就有可能有一个小的搜索引擎，搜索框，或者是在搜索引擎就是通过一些高级的查询语言来搜索，对吧？嗯，明白了，那下面就是想了解一下你作为一个开发者，对于文档体验的期待。就是首先想问一下你写过技术文档吗？

【答】

写过，不过都是那种内部的。

【问】

内部的是给自己看或者是给同组员的人看的是吗？

【答】

使得，因为会有相互协作，就是你写的模块别人可能会来用，就是为了让别人快速的明白你写这个东西的时候是怎么想的，或者是他的思路啊，也不是思路，就是实现的方法以及，但实现方法可能是次要的，更多的是别人会去直接用的那些东西，比如说就是接口参数。

【问】

嗯，你能够大概的讲个你写过的文档的例子吗？然后讲一下这个文档包含哪些内容，然后是给谁看的，以及你是怎么写的，你觉得在写的过程中遇到的难点是在哪里？

【答】

嗯，就拿刚刚说的这个内部的，我是写了一个数据库的一个，嗯，相当一个数据结构的接口，它这个接口提供了有set ，get这些，就比如set就是把一个只给存到数据库里面，get就是把这个旨在从数据库里面取出来，但是可能不是简单的写入取出这么简单的一个操作，可能这个值还会有国际时间啊之类的等等，有一些额外的属性，这些属性都会体现在这些函数的参数里面，所以在我们那个文档里面就是首先就是列举了每一个函数的参数，参数的数据类型以及这个参数的解释。这是文档的主体部分。当时那个文档好像我是分了几个章节，第一开始是第1段，可能也不叫章节吧，因为总体比较短，第一大段就是介绍这个我写的这个数据结构的他是个什么东西，因为是一个比较偏数学上的概念，就是会介绍一下这个数据结构。能起到什么样的作用，然后它对应的论文是什么，当然只是放一个链接上去，这个大家也一般不会去看，这是作为一个参考对，然后第2章是，第1章是这种概述，第2章可能是就是它对于我们，就因为是团队内部，它对于我们内部的这个软件能起到什么样的帮助，就能起到什么样的作用，我为什么要把这个东西给放到我们的这个，就是为什么要开发这个模块，它能带来的好处是什么，然后第3段开始可能就是正文列举每一个参数的解释，那么第4大段可能是一些简单的测试，比如说，会附上一些小的demo，就是一些最简单的demo，就比如这个函数的具体的调用参数，会写一个实例上去。第五大段是一些性能测试，比如说他能在什么样的环境下能跑到多高的QPS，就是它能支持的访问量是多少，这样子，只要是这五部分。

【问】

就是在写的时候相当于你还是分了一下这个文档的框架，那你觉得写的过程遇到什么难点，比较难的问题吗？

【答】

我自己写的文档也不多，可能一开始写就是，首先就是不知道该该写哪些东西，哪些东西该写，有哪些东西不该，因为文档是给别人看的，别人可能并不会关心你这个里面真正的是用的什么思路，用了什么算法，非常具体的一些细节，人家看文档的人可能不关心，所以会需要有一个主次的这样的区分，在自己对写文档不是很熟悉的情况下，就是这个主次可能不是很好的区分，其实我也是看了其他人写的文档，就是翻了一些其他人写的文档，然后大概是凭自己的喜好和感受完，然后再决定写刚才说的那5个章节。主要就是这样，还有就是一些规范规范方面的，就比如在列举那个函数参数以及解释的时候就是会有一些规范，你不是说一大段话就是非常口语的，说一大段话，第1个参数是什么，然后写了一段文字，还有一些规范化的方式，可能是会列一个表格，表格的第1列是函数的参数，就是参数名，第2列是参数类型，第3类是它的具体解释或者是其他备注，就会有一种结构化规范化的方式来表达。

【问】

就是从语言还有从格式上都要结构化，规范化。下1个问题是，就是我们因为要提高这个文档的质量嘛，然后想了解一下开发者对于文档的需求，那你觉得什么样的文档才是好的技术文档，你可以大概的总结一下你能想到的一些点。或者是你期待用到什么样的文档？

【问】

嗯，首先一个就是我刚才也说了好多次，就是那个quick start，其实还是蛮重要的。就是让你在不熟悉这个软件的情况下，能够快速地先跑起来，就是知道他总体上是就是怎样可以跑起来，它需要哪些东西，比如说那个C++语言的话，就是他需要有哪些头文件，然后，就是比如说一个数据库，首先这个数据库使用它有一些固定的模式，就是第1步要打开，第2步要传入一些option，参数就是，相当于你要对这个数据库将要进行一些什么操作，可以先提前传给他一些属性这样子，就是会有一些这样一个固定的模式，他会在quick start里面，文档的quick start里面就会说得很清楚，这样以后你就知道，你在用这个东西的时候就需要先怎么样再怎么样，就这些都是固定不变的。在quick start里面就是让一个开发者能够有一个快速的上手的这样一个认知。其次，好的文档还需要，哦，先说规范吧，起码你要让开发者看起来这是一个，就是专业性要比较强，不能有这种什么错别字，还有甚至是表格用用截图这种东西，看起来非常的不正规不专业。这是第2个，第3个就是章节还有标题，就是大标题小标题，然后就是希望能够分的清晰一些。

【问】

文档的组织要有清晰的逻辑结构，然后格式的话，最好就是要有固定的那种规范，比如说大标题小标题也有规范，最好是用轻量级的标记语言，像markdown来写。

【答】

对，一般这种用markdown写出来的标题会自动生成一个导航，导航就是有，就是它是多级的那种导航，所以可能点开最高级的导航，它下面会有很多小标题，这样子你去按导航去找自己想要的东西的时候，就能够非常清晰的有一个一级标题二级标题这样一个路径，你能够很快的找到自己想要的页面。

【问】

这其实涉及到了一些交互上的设计。

【答】

哦，是的，哦，还有就是跟导航有些相关吧，就是同样的内容不要这里出现一块那里出现一块，因为在分标题的时候，可能因为软件内部结构非常的复杂，可能同一个模块它会和其他多个模块有这种交互，所以可能就会导致同样的一块内容会出现在文档的多个章节里面。

【问】

你觉得这样是不太好的，对吧？

【答】

对确实是不太好，因为你如果想全面的了解这个一个模块的话，可能就要去翻看多个页面，但是有时候这种就是无法避免的，因为软件内部的结构可能会非常的复杂，可能会无法避免，然后这种情况下就是有些文档做的好的就会他会有那种双向链接，就是从这个文档，或者是你鼠标放到这个链接上面，它会有一个小的弹框出来就是能让你大概浏览这个，大概浏览另外一个页面的一些部分，而不用去点开这个页面，因为你点开这个页面可能就你就会跳转到另外一个地方，然后你看文档的这个思路的延续性就被打断了，然后会有这种双向链接这种小的弹框这种出来就是一个比较好的点吧。

【问】

这个框里面就是，这个链接指向的那篇文档的一些大致的比较关键的信息？

【答】

比如说有一个模块，它主要是在页面A里面去介绍的，这个模块又会在另外一个页面里面，可能会比如说用另外一个模块会用到了这个模块里面的一些核心功能，这样的话我想要的这个模块就可能它的主体部分出现在文档a里面，但是文档b里面可能会有一小段也是跟这个模块相关的，这样在文档a里面，可在文档a的最底部，可能会说在模块b，就会有一个备注说在模块b里面，或者是在另外一个模块里面也用到了这个模块的一些功能，然后后面会附一个链接什么，比如说点击此处什么的，然后你把鼠标放到点击此处上面，不需要去点那个链接，他会去弹出一个小的词条，这个小的框框里面的内容就是里面b里面的那一段，而不是整个页面，这样是会让你先知道这个东西，然后你想去详细了解的话，再去点这个链接，会打开页面这样子。

【问】

除了这个之外，你觉得还有哪些比较好的设计的点吗？

【答】

可能还是就是要，页面要搞得好看一点。就相当于有现代化的这种UI的风格，才会看起来会舒服一点，但是这个可能不是特别的重要，可能暂时想不到其他的了。

【问】

就刚才说的这个主要是涉及了三个方面，一个是内容，一个是那个结构，还有交互的一些设计，嗯，那我我们其实刚才也提到一些国内国际的问题，你觉得设置那种语言的切换，这个是必要的吗？

【答】

也是挺有必要的，这个东西，因为中国开发者越来越多，然后在国际的这些社区上也越来越活跃，所以现在很多比较大的软件都会有中文界面，一般都是在网页右上角，什么都会有一个小箭头，或者是会有一些国旗的图标，可以快速的切换语言。

【问】

你认为这个是比较重要的？

【答】

也算是比较重要吧，可能就没那么重要，但是如果有这个，如果他有的话还是会比较好的，算是锦上添花的一个东西。

【问】

嗯，好这个是语言的问题，那你有遇到过那种版本问题吗？就是文档的版本然后跟这个软件或者这个工具实际的版本不对应，文档的版本更新的不及时，然后导致你在用这个文档来做开发的时候，其实本身就会导致一些错误的，有遇到过这种问题吗？

【答】

可能遇到也比较少吧，就没什么深刻的印象。确实是会存在这种问题的，像一般如果比较好的文档它就会在推出新版本的时候，文档会非常及时的更新。然后文档的页面可能他本身也是分版本的，就可能你点开文档，它默认展示的是最新版，但是也有一个下拉框会可以让你选择旧版这样的功能。

【问】

就是你遇到过的文档，它的版本更新还是比较及时的，对吧？

【答】

是的，我觉得比较好的软件配套的文档一般更新都会是比较及时的，当然他们的，如果文档本身是英文的，但是对应的翻译可能更新可能会不是很及时，不过这种情况下一般他会比如说英文更新了，但中文没更新，这样中文页面上他会弹出一个提示框，就会有一个警示框，会提示当前文档不是最新的，就是会提示起到一个提示的作用。

【问】

哦，那你觉得就是现在国内这种做的不太好的文档，他们要改进的话，除了刚才提到了本身的问题，还有其他的可以改进的地方吗？比如说增加反馈，然后让使用者让开发者来给他们反馈，提供这么一个反馈的途径，你觉得这种有必要吗？

【答】

嗯，有必要是有必要，但是一般开源的软件它本身，就是开源的电子化的产品本身就有issue，本身就有反馈的功能。但是反馈的效果怎么样这个还可能真不太好说，可能还是跟国内外的这个开发环境有很大的关系，国内在开源这一块还是做的可能不是很好，所以导致他们的文档质量会相对的低，当然也不是也不是所有的，国内也有像，当然，因为我之前做数据库，对数据库这块有了解，国内也有那个XXXX公司做的那个XXX，对这些东西的文档还是非常的好的。但是像什么XX、XX这些文档就不太行。

【问】

你接触过XXXX的哪一类的文档？

【答】

像XX这边主要还是，比如Paxos Store，也是数据库，XX这边还有一些，主要还是就是Paxos Store，这是一个算法吧，是一个库。但他们的这些开源说实话就像是我为了完成内部的KPI而做的开源，就是可能他就开源一下，然后有很长时间不更新，就像我看那个Paxos Store，好像就上次更新还停留在18年还是17年，然后那个那个issue也积攒了很长，但是开发者也没有回复，就是相当于进行这个开源，你说它开源了吗，它确实是开源了，但是他又停止维护了，开发者也不去回复，相当于是一个半死的这样一个状态。

【问】

就是即使是提供了这种开源或者是提供了反馈的途径，他们也没有好好地去回应这个反馈？

【答】

对，有途径，有反馈，但是就相当于是不管的这样一种状态。

【问】

那就除了这个之外，你还能想到其他的吗？如果说要你给他们提供改进的建议的话，还能想到其他的点吗？对了，刚才提到这个国内国外开发的生态，你觉得啊，国内的开发生态不太好，那除了开源这一块，还有其他方面可以讲吗？就是说你觉得国内的这种开发生态不太好，一是开源做的不太好，然后还有其他做的不太好的吗？

【答】

要和文档有关系吗？

【问】

对，和文档有关系。

【答】

如果不是开源的话，就是闭源的软件的文档可能就主要是面向内部，当然这些质量不好，我觉得也是非常正常的，国外的，但我也没有了解国外的这些内部文档是怎么样，我估计都好不到哪里去。因为一是面向内部的一些文档，本身它也不算指标啊，也不算你的业绩什么的，所以大家可能就只要有就行了，也不会去非常认真的去写，然后可能写完以后也就很久不会去维护，对像我之前实习就有些文档已经非常旧了，就是，我还去问过别人就说，唉，你这个东西怎么用起来和文档不一样，他说，这个东西就是已经更新过很多轮了，已经不是文档里那个样子。

【问】

就是说文档的更新赶不上软件的迭代的速度。

【答】

对，所以还是希望公司内部能对文档，我希望能越来越重视。就靠自觉不行的话可能可以纳入一些指标啊，纳入一些就是考核的指标，但是这样可能也不太好，这是一个想法，就是希望能够让开发者有一些动力去主动的维护他们这种这些文档。

【问】

其实现在在公司内部他写文档的这个团队也不全是开发者，还有那种就专门写作基础写作的人员，这些人可能是有文科的这种写作背景，就是他们是一个混合的团队。

【答】

这可能不是很了解。

【问】

嗯，对，因为就是开发者本身他可能这个写文档的意愿不是很强，比如说你作为一个开发者你愿意花时间来写代码可能可以接受，但是花时间来写文档可能是不太能接受的。

【答】

我就想到一个笑话吧，说程序员最讨厌两种人，一种是，不是，是最讨厌两件事，一种是别人不写文档，一种是自己要去写文档，就是会这样。看到别人的东西没有文档就会很恼火，但是到自己写文档的时候，自己也不想去写。

【问】

还能想到其他的点吗？还有需要补充的吗？对于文档的这个期待，文档可以改进的建议。

【答】

嗯，可能刚才说的也比较全，就是也主要想到的是这些，还是希望文档，就文档这个东西对于开发者来说真的非常重要，但是确实大家都不愿意去写，这可能需要一些什么样的方法，当然可能是更多的管理上面的一些方法去让文档这一块越来越好吧。

【问】

好的没有其他可以补充的对吧？行那我们的问题就问完了，今天就先到这儿。