*问：想大略了解一下您的做开发的经历，您大概有过多长时间的开发的工作经历？*

答：正式开始工作应该是2015年四五月份。

*问：我们想了解一下开发者的特征，因为有可能开发者他们本身的一些习惯和学习习惯等等工作的习惯都会影响到他们对于文档的使用，所以我们就想了解一下这个开发者，您觉得程序员这个群体和其他专业其他职业的人相比有哪些不一样的地方。您可以大概从学习啊，还有工作生活的方式，习惯特点直接来简单说一下。*

答：感觉有一点就是程序员得一直学，因为这有一些技术知识更新换代是比较快的。然后另外程序员因为跟电脑打交道比较多，可能很多程序员可能相对来说可能会比较内向一点

*问：学习的方式有什么特点？*

答：一部分是看文档，另外一部分还是直接的操作，就是有很多书会教大家；但高效的学习方式是去真正的实际操作，比如说照着代码去实现。

*问：相当于是就是边学知识，然后边实现，就是边学边做？*

答：对，而不是单纯的去学习理论知识。

*问：最近或者是之前有学过一项什么新的软件，新的工具或新的技术吗，或者某个框架之类的，然后您可以大概的回忆一下整个学习的过程是什么样的呢？可以举个例子，具体的例子。*

答：我主要做容器这块，比如要学习一个框架的话，需要看它的介绍，看是否能满足我们的需求。实际工作中大多数还是问题驱动去学习，会去先了解这个项目的一个大概的一个情况，在真正的深入去学之前，你要先去了解它的整体；比如通过**FAQ**了解一些局限性、和同类产品的对比，了解下它的优缺点，然后最后再决定我们是否值得去花时间去深入的去学习

*问：那你了解一个产品就是通过他的这个官方的介绍文章是吗？*

答：会从侧面去了解产品与同类竞品的对比

*问：怎么了解呢？*

答：一般通过搜索引擎。

*问：前期入门一项产品时主要是了解，然后中期后期是怎么来学这个东西呢？怎么学、怎么*用？

答：首先肯定是先看一下这个东西大概怎么跑起来，架构上的一些东西啊，就是它的原理性的东西可能在最开始调研阶段就已经知道了，然后真正要上手去用的话，我可能需要去看一下他大概是怎么跑起来的；如果以代码形式集成到我们的项目里的话，需**要去看一些代码示例（sample），比如照着命令大致走一下quickstart**，这样就能够大致入门；然后根据我的需求的实现，**针对性去查他的那个对应的章节的文档。**

*问：此时相当于是开始用起来了。那您用的比较熟练的时候，比如说后期又会继续使用这个文档。追：后期可能会有遇到一些问题，就比如说您遇到问题的时候，你是怎么办呢？*

答：会先看一下文档，如果找不到内容，会使用站内搜索，有些网站会提供这个功能；搜不到再去用谷歌搜索。如果是一个开源项目，会去github上搜索，因为可以直接搜到相关的代码；比如一个报错提示，可以**直接在github上搜索查看相关代码**，因为可能先前已经有人提了issue，然后开发者社区内已经有人尝试解答了问题，可供参考。

*问：可以举一些用搜索引擎搜索错误信息的例子吗？追：一般会用搜索引擎搜索什么样的问题？*

答：一般是报错的信息。

*问：一些比较系统性的，宏观的问题呢？*

答：比如如何达到某种想要的效果，就会使用how to 之类的去搜索。

*问：追：就是说构建的****表达式****是how to ？*

答：有时可以去掉how to，直接搜关键词。

*问：您刚才提到了站内搜索？*

答：站内搜索用得不太多，因为很多**站内搜索实现效果不佳**。会去看**目录**，大概看每个章节讲了什么内容。

*问：追：也就是按照章节来定位。技术文档，在您的开发工作当中扮演什么样的角色？*

答：最开始的时候，入门的时候会去用的会比较多一些，然后后面实际，因为文档其实它也分多种，比如**API文档**，写代码的时候可能真正会去参考。有些ide，就是它可以支持直接在写代码的时候给你提示，然后包括提示里面也有那个文档。API文档，其实是在真正写的时候用的会比较多一些，但可能是直接跳转到对应的源码上面，很多编程语言支持通过源码里的注释生成此类文档。（追问：**自动化生成文档**）

*问：开发过程中使用文档主要还是为了查询一些参数对吧。除了API还用过哪些类型文档？*

答：项目提供的一些博客性质的文章，偏重说明解释，比较容易读懂；API其实就是你深入到功能实现的时候才会去用，不会去通读。其他的我这个用的会比较少吧，就是最开始可能就是标准的文档里面看一下它的一些**介绍，然后入门**。

*问：会看FAQ文档吗？*

答：在最开始就会看，因为一般都会比较短，就是基本上就是一页，再就是去研究这个项目的时候，就会了解它到底有哪些问题

*问：一般通过什么方式来找到需要的技术文档*

答：直接前往github项目页面，或是产品官网

*问：也会使用搜索引擎？*

答：产品大多会把官网放在搜索结果最开头。

*问：所以搜索引擎是寻找官网入口，所以您找这个技术文档的途径其实就是前往官网？*

答：有访问记录之后，后续就是基本上Google都不用了。如果对一个项目比较熟的话，其实你就大概给他的那个结构甚至URL就能差不多能记得住。

*问：遇到问题想要通过谷歌搜索，是如何搜的？【提问重复】*

答：直接搜错误信息、如何做一个事情，实现一个功能。

*问：用自然语言来说或者关键词来搜索？*

答：**最开始会去搜产品和其他类似项目的对比。**

*问：在哪里能够看到这类内容？*

答：有些项目，他会自己去介绍跟同类产品的对比。

*问：就在官网里面吗？*

答：github项目页面和官网其实差不多。有些github项目可能不是在正式文档里，而是在issue里去找。

*问：阅读文档的时候，您是怎么阅读的呢？如何找到如何快速的定位自己需要的信息？*

答：先了解概要，如果有目录大概扫一眼，根据目录导航定位内容，快速查看。站内导航用得不多。

*问：您应该也看过一些国内的国外的挺多的文档，那你可以回想起一些比较好的文章的例子，然后说一下他们好在那里，就用起来比较顺手的。*

答：编程语言的话，go语言和rust语言做的都不错，API文档里会包含一些example；rust语言的API文档还会对功能做一些原理性的解释。

*问：还有其他哪些比较好的点？刚才说的都是国外的，你可以说像也可以说像国内的一些。*

答：没怎么看过国内的文档，一般看英文。很多文档其实也有中文，但是因为中文有时候跟那个英文还不太一样，所以有时候而且他那个翻译后你都看不出来它的原词到底是什么。可能是习惯了英文的表达，一下子切回中文之后不够直观。

*问：英文文档读的比较多，是否有阅读困难？*

答：中文阅读更顺畅，可以快速抓住主要信息，英文阅读很难达到同样效果。

*问：还是能够读懂？*

答：对，但是会慢一些。

*问：那您可以举一些比较失败的文档的例子，就写的比较差的，你见过吗？不太好用的。*

答：【思考时间长】静态编程语言需要用类型来标明API文档，自身的说明比较少。

*问：追：内容不够详细？*

答：对，说了和没说一样。

*问：追：还有其他的吗，也可以不用举一个某一个具体的例子。*

答：描述的不清楚，就是基本上没有什么太多的东西，只让你看代码。还有就是文档内容过时，和实际操作有出入。

*问：追：按照文档说明无法得到相同结果？*

答：对。

*问：您有写过技术文档吗【建议改进问法】*

答：怎样算技术文档？【概念不清晰】

*问：内部的也可以算，比如说内部的项目之间的沟通，比如设计文档之类。*

答：肯定有。

*问：难点在哪里？*

答：撰写还好，主要是维护。你写个东西，最后发现他很快就过时了，然后因为写文档不是说马上就能把这个整个给描述清楚，然后你花了一些时间就会发现很快就过时了，这个维护成本其实挺高的，然后而且我觉得，程序员这个很多不爱写文档的人，可能有一方面是这个原因吗？就是你写了之后发现内容变了又得重新写。

*问： 您觉得是否很多程序员都不爱写文档？*

答：对，大部分是这样。

*问：是否是因为觉得写文档没有太多效益或是成就感？*

答：倒不是因为这个。一方面表达能力可能稍微欠缺，他都认为自己可能感觉有些东西表达的不是特别清楚；另一方面，公司内部的一些项目是我们在开发的时候，其实可能这个项目迭代很快，然后这个你写的可能很快就过时了。

*问：怎样的文档是好文档，期待用到怎样的文档，您可以大概的总结几个文档设计的维度，说一下那么好的设计的点。*

答：首先我觉得文档就是还是要有结构的，就是他的那个层次结构，然后嗯，就是我觉得其实那个一个完整的文档网站，那就跟各篇文章一样，他就是在这种结构就是最开始或者他的一些，啊会比如说明他她是啥东西，然后介绍简单介绍它的一些原理架构，然后他的一些FAQ都能够提前的去看到了。关键信息出来之后之后，然后再深入去看那些文档，然后并且一些API文档的话，最好是要有这种**示例**。因为刚开始上手的时候，你可能很多时候你不清楚它到底怎么跑起来，到这个时候就需要包括那些参数应该怎么设置。

*问：就从内容上说是要提供比较具体的事例，可以跑起来的，是结构上的话，就是有一个比较清晰的组织结构，那您可以大概总结一下，您看过的文档它的结构都是什么样子的，就是第1个第2部分是什么，它有规律吗？*

答：其实大部分都还算比较有规律吧，就是至少就是经过很好的组织的那种，就是一些比如说刚开始会介绍他这个项目大概是什么，使用在哪些场景下？然后它的架构大概是什么样子的，然后工作原理是什么，FAQ一般在最后，quickstart在最前面。现在一些很多编程语言，它其实都直接带一个在线的一个demo，在线就可以运行，在线的去看到他的结果，这种比较直观。

*问：还有其他要点吗？*

答：【思考时间长】展现形式、格式之类的。

*问：排版格式需要比较舒适？*

答：页面特别宽的那种，就是像就会比较费劲。

*问：后面一个问题是关于文档的反馈的，那您见过的文档里面，它会有在文档的末尾或者是*某个其他的地方给您提供一个反馈的途径呢，这是什么样的反馈？

答：基本没有点击过。比如点赞、反馈等。基本没有反馈过。

*问：为什么没有反馈呢*

答：可能是懒吧【有待深入挖掘】。

*问：最后一个问题就是关于文档交互的这个关于这个文档是这个界面的交互的设计没有，有些什么样的期待，或者说您看到过哪些比较好的交互设计？*

答：一些代码片段要能够比较方便地去复制、代码可以线上试运行、提供一些能够放大查看大图的功能架构图、锚点，也就是链接，看了文档之后你可能会发给其他人去看，按照锚点定位其他人直接就跳过去就能看到、UI方面包括提供暗黑模式。