**一、专业技能：**

这些【专业技能】都写的很少，其实是可以在丰富一些的。

我来给大家拓展一下、

* 熟练C++，（列举C++的若干知识点），了解 Java，python，go （适当补充对这些语言的理解）
* 熟悉常见设计模式（例句一些设计模式）
* 熟悉linux操作系统vim开发环境，（列举网络编程相关知识，例如epoll，socket等等）
* 熟悉网络，（列举网络协议相关考点，tcp/ip,http, https, 三次，四次握手，流量控制等等）
* 数量掌握数据结构与算法（列举常用算法，最好搞透一个算法，说对该算法有独到见解）
* 数量使用Git，等版本控制
* 以上为公共写法，下面可以在补充自己的其他领域的内容

针对以上这个模板， 再来补充相关内容： （可以缩小面试官提问范围，把问题具体一点，你也可以有针对性的准备）

* 熟悉C/C++，熟练使用C的指针应用及内存管理，C++的封装继承多态，STL常用容器，C++11常用特性(智能指针等) ，了解 Python，Gtest等。
* 熟悉常用设计模式(单例模式，工厂模式等)
* 熟悉Linux下vim开发环境，了解网络编程，IO多路复用，epoll等等。
* 熟悉OSI五层网络模型，熟悉TCP/IP,UDP,HTTP/HTTPS,DNS等网络协议，熟悉TCP三次握手，四次挥手，流量控制，拥塞控制等手段。
* 熟悉操作系统的进程通信、死锁、内存管理等知识。
* 熟练使用 MySQL，熟悉 MySQL 索引、事务、存储引擎、锁机制。
* 熟悉常用的数据结构(链表、栈、队列、二叉树等)，熟练使用排序，贪心，动态规划等算法。
* 熟悉使用Git，vscode工具使用。

但需要注意的是，这里写的点，自己一定要熟练掌握，因为简历上写的，面试官一定会问。

这样有一个好处，就是 **缩小面试官的问题范围**， 只要简历上写的，你都准备好了，那么简历上的知识点面试官一定会问，这样你就掌握了主动权。

举一个例子，如果简历上直写：熟悉C++。其他都没介绍，那么面试官指定围绕C++漫天遍野的问起来了，你也猜不透面试官想问啥。

如果简历写熟悉C/C++，熟练使用C的指针应用及内存管理，C++的封装继承多态，STL常用容器，C++11常用特性(智能指针等)。那么面试官基本上只会问，内存管理，多态，STL和C++11的一些特性， **这样你就把面试官的问题都圈在可控范围内**，从而掌握主动权！

这一点很重要，希望大家要有这个思路，去写自己的简历。

**java 版本**

不少同学是java的，这里我在给出一个java的版本。

* 熟练掌握Java基础，集合等相关知识，了解常见的设计模式。
* 熟悉JVM的垃圾回收机制、类加载机制及Java的内存区域。
* 熟悉Java并发编程，掌握JUC中常用的工具类，如ConcurrentHashMap等，熟悉多线程，线程池，Java内存模型。
* 熟悉OSI七层模型和TCP/IP四层体系分层结构，掌握常见网络协议，如HTTP/HTTPS 、TCP、UDP、DNS等。
* 熟练使用 MySQL，熟悉 MySQL 索引、事务、存储引擎、锁机制。
* 熟悉操作系统的进程通信、死锁、内存管理等知识。
* 熟悉Redis数据类型使用场景和内部实现，熟悉持久化和过期淘汰策略，熟悉缓存高并发场景，比如缓存穿透、缓存击穿、缓存雪崩。
* 熟练使用Spring Boot、Spring、Mybatis等常用框架，熟悉 Spring IOC 、AOP 原理，了解 Nacos、Zookeeper 等常见组件。
* 掌握 Linux 常用命令，如 netstat 、grep、top、chmod、find 等。

**前端写法**

* 熟练掌握 HTML、CSS 并能精确还原设计稿
* 熟悉 HTML5 和 CSS3，熟练使用 CSS3 弹性布局等，能配合媒体查询实现响应式布局、自适应布局
* 熟练掌握 JavaScript，理解作用域、闭包、构造函数、事件委托、异步加载等
* 熟悉 ES6 基础语法、箭头函数以及 ES6 模块化语法，了解 promise
* 熟悉 JQuery、Bootstrap 等前端开发框架进行前端项目开发
* 熟悉 less，熟练使用 less 语法
* 熟悉 Vue、Vuex、Vue-Router、Vue-Cli 等 vue 相关开发技术
* 熟悉 React、React-dom、React-router、Redux 等相关开发技术，熟悉 dva 数据流方案
* 熟悉 Echarts 等可视化工具
* 了解 node.js 及 koa2 框架，能和前端实现简单数据交互

**go后端写法**

* 熟练掌握GO语言，了解map， struct， slice， channal，GMP模型调度器，gc垃圾回收，内存逃逸等底层原理；
* 了解掌握go-micro，gin，grom， grpc，protoc等框架使用；了解Python，Java等其他编程语言
* 熟悉计算机网络原理，了解TCP，UDP，HTTP，HTTPS等网络基础知识
* 熟悉使用MySQL数据库（InnoDB，MyISAM储存引擎）及数据库管理工具Navicat，熟悉InnnoDB储存引擎 的底层框架及MVCC多版本并发控制
* 了解Redis等非关系型数据库
* 掌握操作系统进程，虚拟内存，锁，调度算法，多核缓存一致性等概念
* 掌握IO复用并发模型，Linux系统命令及云服务器项目部署算法相关。
* 熟悉基础数据结构，掌握多种排序，动态规划，二叉树等算法
* 微服务相关：了解基础微服务架构，服务网格架构
* 团队相关：熟悉使用Git以及使用Github／Gitlab仓库进行协作开发

**算法岗**

算法岗更强调的是对 机器学习那一套的学习。但如果 对计算机基础也了解的话，也好也写上，以下我就列一列 计算机学习相关的。

* 熟悉python编程
* 熟悉基本的机器学习算法（SVM、决策树、K均值聚类等）
* 熟悉基本的图像处理算法、熟悉使用OpenCv的基本算法
* 熟练掌握深度学习框架PyTorch,了解TensorFlow
* 熟悉深度学习目标识别算法原理（Faster R-CNN、Cascade R-CNN、DetectoRS、YOLO系列）以及深度学习语义分割
* 算法原理（UNet、DeepLabv3+系列）

**测开岗**

* 熟悉 linux 常用命令,如：文件操作、权限管理、进程等
* 熟悉 c/c++、python 编程语言,了解 stl 常用容器,熟悉面向对象模式
* 熟悉 mysql 数据库,了解索引、事务、隔离级别 熟悉计算机网络知识,
* 了解 tcp/udp、http\https 等基本的网络协议
* 了解 postman,fiddler 软件测试日常工具的使用
* 掌握软件测试基础理论,测试用例编写方法
* 熟悉自动化测试技术，以及自动测试工具 QTP 的使用；
* 熟悉性能测试技术，以及性能测试工具 loadrunner 的使用；

**大数据**

* （Java基本技术栈和计算机基础，参考上面Java后端的写法）

• 了解 hadoop 生态圈相关组件（hdfs、yarn、mapreduce 编程模型）。

• 了解 kafka 消息系统，了解 kafka 架构组成、存储机制、生产者消费者相关内容。

• 了解 spark 计算框架，熟悉 RDD 算子的使用，了解 spark shuffle 机制、数据倾斜、作业提交流程等

**Android**

1.有扎实的 Android 开发基础，包括生命周期，事件传递机制，自定义 view，动画等；

2.熟练掌握 Android 消息机制 Handler，及其相关源码实现；

3.熟练掌握 Android 进程间通信的各种实现，包括Bundle、Socket、Binder、文件共享；

4.熟练掌握 Android 系统的启动过程，应用的启动过程；

5.熟练掌握 Android 系统服务，包括AMS，PMS，WMS等；

6.熟悉 EventBus、Glide、LeakCanary 等三方库的源码；

7.熟悉 Android 应用性能优化，包括内存，GPU，网络等方面；

8.熟悉 Java 数据结构的源码实现，比如ArrayList、HashMap；

9.熟悉 JVM 的 GC 机制、类加载机制及内存区域；

10.熟练掌握 TCP/IP 体系分层结构，熟悉常用网络协议，如HTTP/HTTPS，TCP，UDP 等；

11.熟练掌握 Flutter 开发，理解 Flutter 中的Widget tree、Element tree 等基础概念；

12.熟练掌握 Flutter 混合 Android 开发，有插件开发经验；

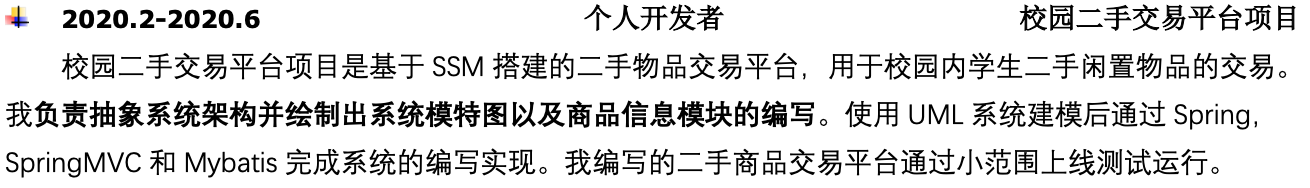
**二、项目经验**

**简历中【项目经验】这么写**

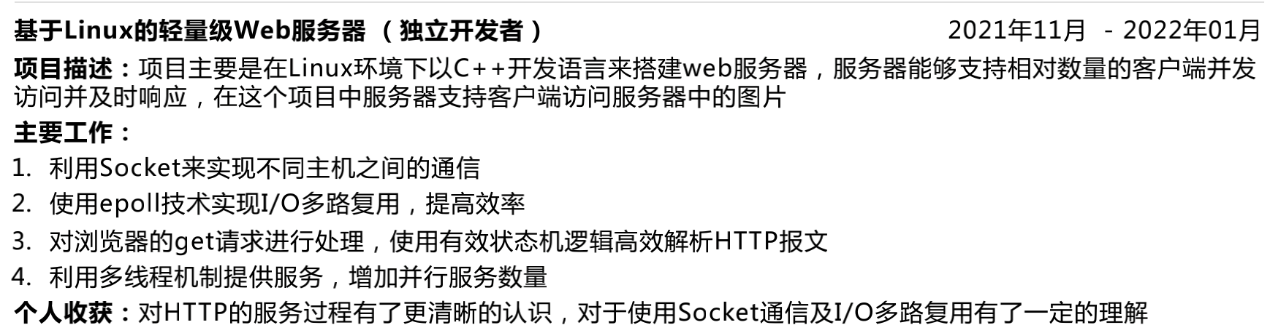
本文我给出，C++和java 以及前端项目的具体写法，其他语言的话，也可以参考：

看了一位录友的简历。

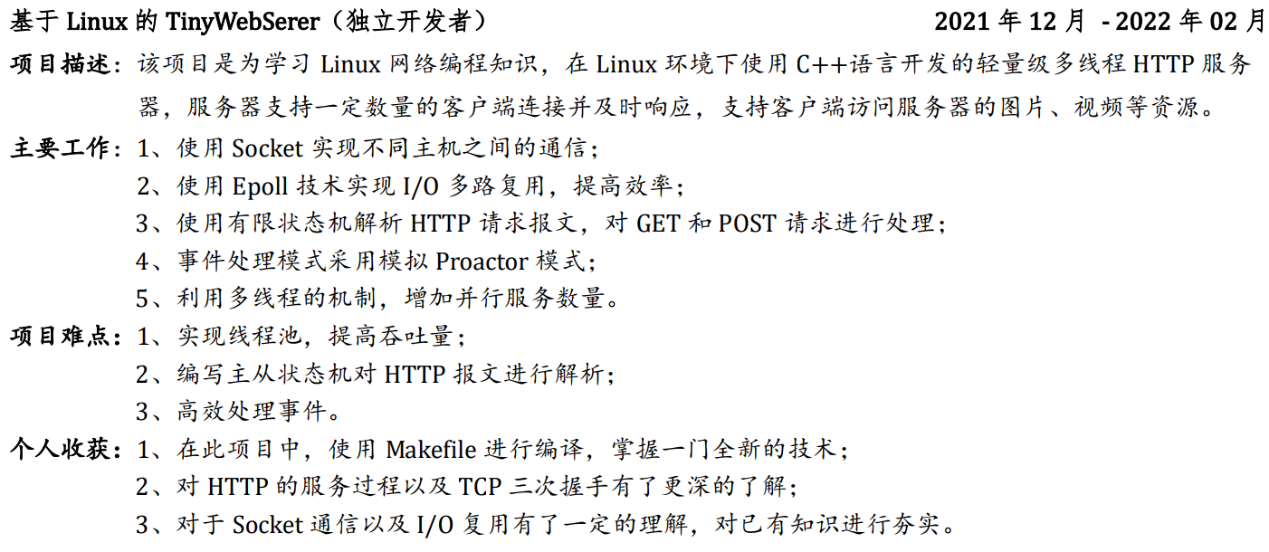
【项目经验】都堆在一起了。



建议【项目经验】分 「项目描述」「个人工作」「个人收获」这三块来写，不要堆在一起。例如这样：



或者在写一写「项目难点」例如这样：



至于「技术栈」可写可不写。例如有的同学在【项目经验】中把技术栈都写出来了，例如这样：



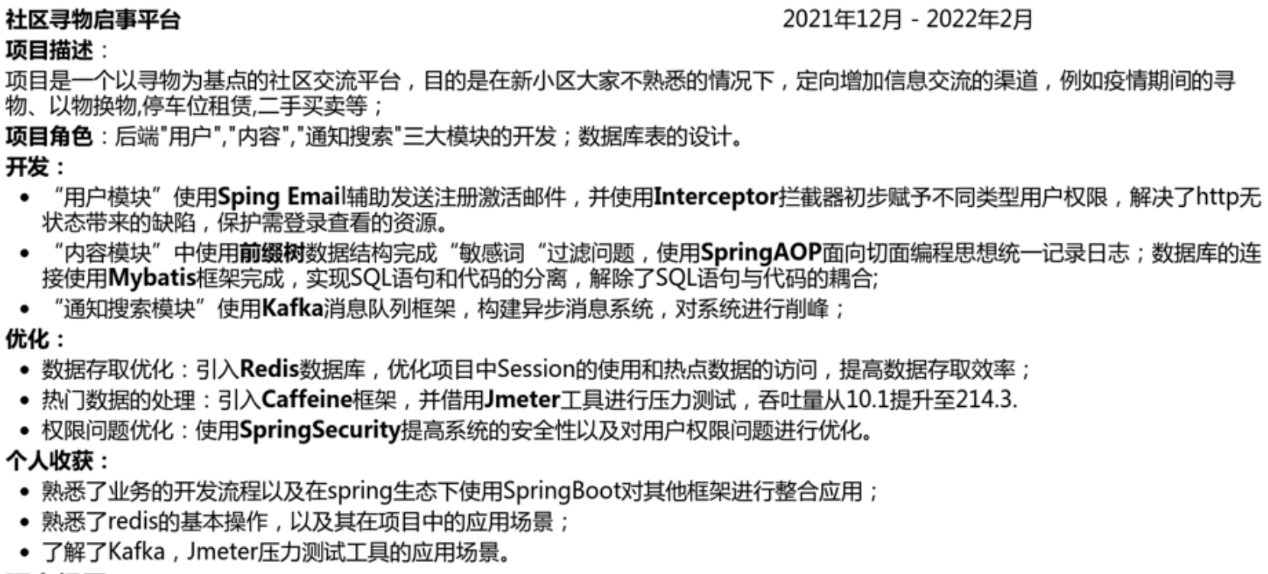
这样写的好处是让面试官快速了解，你这个项目的技术栈。缺点是列出一大堆技术，怕面试官详选一个细问起来，自己在回答不上来。

不写 「技术栈」的话就在 「个人收获」 写一写自己使用某一技术的心得，这样把面试问题缩小到自己可以把控的点上。

如这样：



论坛项目其他写法：





三、自我评价

一般来说 「自我评价」都是写在简历最后面，如果学历不高，可以在「自我评价」上好好下功夫把「自我评价」放在最前面。

很多录友写「自我评价」是这样的：

有较强的适应能力、学习能力、抗压能力、自我管理能力。工作 态度认真负责，具有良好的团队合作精神，生活态度乐观开朗， 积极向上。爱好跑步、爬山、钓鱼、羽毛球等。喜欢做有挑战的 事，遇强则强，压力越大，激发的能力越大

或者是这样的：

具有较强的学习能力，善于发现和思考问题，对工作认真严谨；乐观向上，具有团队协作能力。

总之就是写了，和没写区别不大。

我给大家列几个点，可以从信息检索能力，动手能力，解决问题能力 等这几方面来说，例如这样：

* 有较强的信息检索能力，擅长坚决疑难杂症，通过Google/Github/StackOverflow等国外论坛/文档解决技术问题。
* 动手能力强，有探索精神，拆过路由器，笔记本，组装过PC，很早就用过XXC，最早的一批XXX用户。尝试使用各个云服务的产品，对象存储，云服务器，CDN 等等。
* 在开发中对待问题具有认真求索的精神，能够在短时间内解决问题并理解知识点。并且自身非常细心，严格要求自己规范性的书写代码，这让我少写了很多 bug。

我也是看到有录友这方面写的不错，给大家分享一下， 当然以上仅仅是一个参考，可以在 「自我评价」中写一方面，写几句话，甚至不写，都是可以的，也取决于自己简历的篇幅。

四、实习经历

匿名用户 提问：

卡哥，秋招面试问实习内容，一般是怎么问呀，需要怎么准备的呢，后端实习，主要负责业务这一块

实习都做了什么。

实习中遇到了哪些问题（技术上，开发流程上，部门沟通上）。 你是如何解决这些问题的。

实习中收获了什么

技术上成长 （学习了哪些技术组件，自己有没有去深挖）

代码风格上 （如何写出易于维护的代码）

开发流程上 （单元检测，打日志，Git团队合作）

对业务理解能力上 （如果把业务场景转为系统设计转为工程代码，都考虑了哪些问题）

也可以适当把同事工作的内容写自己简历上，但前提是自己一定要研究清楚，否则就是给自己挖坑 「实习收获」

模板写法：（不建议原封不动的抄，很容易重复） ，建议根据自己的实习内容，学到的具体技术来写。

模板写法一： 对团队开发的流程有了更加深刻的理解，提升了自身的技术水平，学会了技术文档的编写，锻炼了开发的思维，更加注重程序的可维护性、模块化和可读性，是自己明白了快速学习新知识并将其运动到业务的重要性。

模板写法二： 熟悉了企业项目的开发流程，提升了个人能力，其中包括但不仅限于:快速接受新事物代码规范、良好的开发习惯(如先画整体架构图)、团队协作(如及时沟通)、问题的快速定位等。