**搭建现代的开源软件开发环境**

江西科技师范大学：张梓鑫

**摘要（Abstract）**

通过查阅开源软件的历史背景和有关工具的使用方法的有关文献，结合传统的代码开发工具和代码编译工具，尝试搭建一个现代的web应用程序的开发学习环境。主要利用了linux/unix的命令行(cli)探索软件项目文件和文件夹的管理，再运用git工具初步完成了从github上下载代码仓库，查看仓库的状态以及修改仓库中的代码文件，最后在利用现代浏览器在本机中实现了代码的运行，另外为了让开发效率更高，创造性地实现了修改linux的配置文件，让本人喜欢的编译软件使用简单的键盘命令可以快捷地启动。

**关键字（Keyword）**

linux/unix，命令行(cli)，git,github,GNU，GUI

1. **开源软件历史背景和命令行操作**

目标：深刻理解开源的概念，了解开源软件的历史背景。

1. 什么是开源软件

从最广义来讲，一个软件是开源的，最基本的含义是代码是公开的，任何人都能进行查看，修改，使用。那么这一概念是如何诞生的呢？

1. 开源软件的历史发展

讲述开源软件的起源就不得不谈自由软件（free software），其历史可以追溯到上世纪50年代。1949年5月6日，世界上第一台完整的冯·诺依曼结构计算机在剑桥大学开始运行，这天因此被认为是现代计算机的诞生日。我们在这里因为内容侧重原因不做过多概述，我们直接快进到linux诞生之初，时间来到1991年，自由软件基金会已经开发了GNU操作系统的大部分组件（例如编译器、编辑器、用户界面等），但是仍然未完成作为操作系统核心的内核，即GNU Hurd。时至今日，在自由软件基金会成立的35年后，GNU Hurd仍没有释放出可用的1.0版。这不能怪任何人，因为开发一个操作系统内核的成本实在是太高了。完成操作系统内核的任务是在一个21岁的芬兰大学生Linus Torvalds的带领下完成的。在0.02版后，来自世界各地的程序员们加入到了Linux内核的开发中，使其迅速完善。

在《The Linux Command Line A Complete Introduction》这一书中也有相关的内容：

some years earlier, Richard Stallman began the GNU Project to create a free Unix like operating system.

in 1991, Linus Torvalds wrote the first version of the Linux kernel.[[1]](#footnote-0)

在之后的发展里，Eric等人引入了开源软件（open source software）这一用语，并获得成功。这便是开源软件的由来。

1. 命令行操作

* 命令行是一种通过文本界面输入指令与计算机进行交互的方式。用户可以在命令行中输入特定的命令来执行各种操作，如打开文件、创建文件夹、运行程序等。命令行通常不提供图形化界面，而是通过键盘输入文本命令来完成操作。在大多数操作系统中，如Windows、Mac OS和Linux，都提供了命令行界面来方便用户与计算机进行交互。
* 那么我们为什么要使用命令行，他比起现在的图形化操作界面有什么优点呢？

其实对于专业人士和需要使用电脑进行大量重复且复杂的操 作的时候，GUI界面操作是极其不便和效率低下的，例如我们 要删除几个深层文件夹中的内容。在windows中我们首先要 知道文件夹的位置然后进入再选择进行删除。这听上去似乎 并不是很繁琐，然而当你不知道地址的时候，你就会知道搜 寻这件事是很麻烦的。但这个问题再命令行中是很容易解决 的，因为命令行中集成了很多快捷指令，文件的删除比起 window是相当方便的，只需要输入一些简单的命令就可以完 成操作，对于简单的操作是这样，对于难的一些操作也是这 样，William Shotts也在其书的Why Use the Command Line 部分说过：

It’s been said that “graphical user interfaces make easy tasks easy, while command line interfaces make difficult tasks possible,” and this is still very true today.[[2]](#footnote-1)

这也直接说明了命令行的各种优点。

**二．建立和管理自己的本地项目文件夹**

1. 建立自己个人的照片log文件夹

2.建立自己的项目文件夹

有关命令：

* Cd /d
* mkdir zhangZX
* Cd zhangZX
* Mkdir picture

**三．修改linux系统的profile配置**

目标：实现在命令行（cli）中快速启动代码编辑器。

参考文献：/etc/profile A global configuration script that applies to all users.[[3]](#footnote-2)

* 打开 Git Bash 终端窗口。
* 输入以下命令，将 EditPlus 可执行文件所在目录添加到 PATH 环境变量中：echo 'export PATH="$PATH:/c/Program Files (x86)/EditPlus"' >> ~/.bashrc
* 这个命令会在你的用户主目录下的 .bashrc 文件中添加一行配置，将 EditPlus 可执行文件所在目录添加到了 PATH 环境变量中。当下次打开 Git Bash 终端窗口时，就可以直接使用 editplus 命令来启动 EditPlus 了。
* 关闭当前的 Git Bash 终端窗口，并重新打开一个新的窗口，或者输入以下命令使刚才添加的 PATH 配置生效：source ~/.bashrc
* 现在，你应该可以在 Git Bash 终端窗口中使用 editplus 命令来启动 EditPlus 了。例如，要编辑一个名为 index.html的文件，可以使用以下命令：editplus index.html

**四．下载github上的开源软件代码仓库**

**目标：**用ping命令来理解网络各层次的连接和服务（注意关闭加速器），再用git clone 下载网上仓库https://github.com/masterLijh/masterLjh.github.io

* 用ping探索GitHub网站的连接能力用ping探索github.io代码仓库的连接能力
* 测试GitHub提供的web服务
* 用clone进行代码仓库的下载：输入：git clone <https://github.com/masterLijh/masterLijh.github.io> 或git clone <https://github.com/lijianhong19720904/myCTApp.git按下回车键后开始下载对应代码仓库>
* 用git log查看当前代码仓库的提交日志

**五．在本地测试和运行项目代码**

**目标：**用代码编辑器来查看我们下载的Web Application，并用浏览器来观察代码的运行情况和变化。

可在浏览器中输入masterlijh.github.io（下载了第一个链接）或 打开D:/Git/myCTApp/index.html（下载了第二个链接）进入对应网页

1. The Linux Command Line A Complete Introduction-introduction [↑](#footnote-ref-0)
2. The Linux Command Line A Complete Introduction-introduction [↑](#footnote-ref-1)
3. The Linux Command Line A Complete Introduction-119 [↑](#footnote-ref-2)