**Semana #1: Ferramentas e Introdução a computação**

Dia: 20/04/2020

Ferramentas

* Git
  + <https://git-scm.com/book/en/v2>
* GitHub
  + <https://github.com/>
* Python
  + <https://www.python.org/>
* Editor de texto
  + Sugestão: Sublime
  + <https://www.sublimetext.com/>

Tarefas

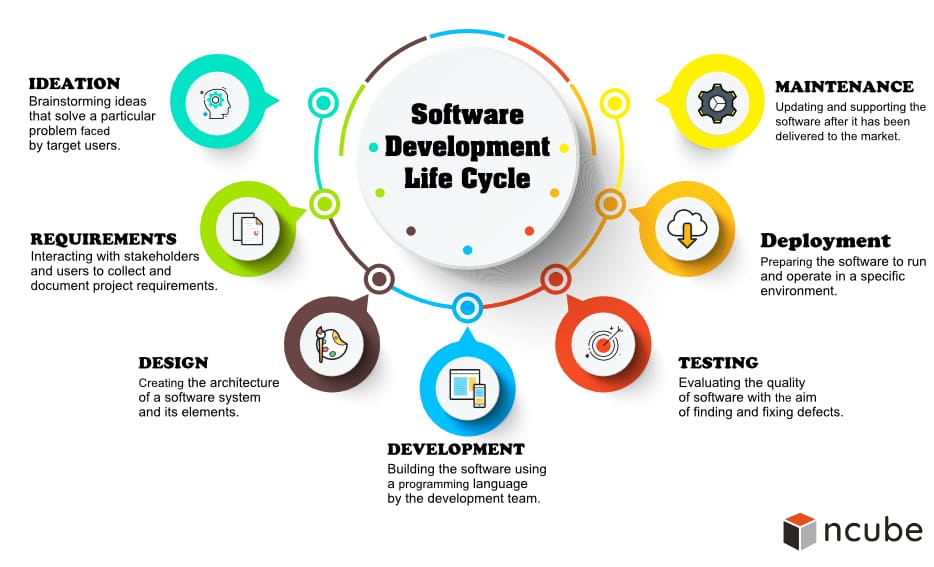
1. Criar conta no GitHub
2. Instalar o GitHub Desktop: <https://desktop.github.com/>
3. Instalar o Git
4. Instalar o Python 3.7+ (qualquer versão acima da 3.7)
5. Instalar um editor de texto para programação
   1. Tem vários. Podem pesquisar. Ouvi falar que o Sublime é muito bom, mas nunca usei. Eu uso o VIM, mas não recomendo pra vocês agora.
6. Abrir o gerenciador de tarefas e ver os processos rodando no seu computador.
7. Estabelecendo uma meta de longo prazo: escrever um parágrafo sobre o que você gostaria de trabalhar em programação e em qual empresa. Jogos? Ecommerce? Mercado financeiro? E onde? Acesse o site da empresa, veja se tem uma lista de vagas de emprego para programador e escolha uma para você.
   1. Renato: <https://www.linkedin.com/jobs/view/1616623569/>
   2. Kaillen:<https://www.linkedin.com/jobs/search/?currentJobId=1782528729&keywords=Ita%C3%BA%20Unibanco>

Textos sugeridos:

1. Git: <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>
2. Git: <https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git>
3. Git: <https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F>
4. Pull Request: <https://help.github.com/en/github/collaborating-with-issues-and-pull-requests/about-pull-requests>
5. Processos (apenas o item 1): <https://www.computerhope.com/jargon/p/process.htm>
6. PIP: (não precisa ler inteiro) <https://realpython.com/what-is-pip/>
7. Django: <https://tutorial.djangogirls.org/en/django/>
8. Interview: <https://www.roberthalf.com/blog/evaluating-job-candidates/16-web-developer-interview-questions-every-hiring-manager-should-ask>
9. Interview: <https://www.techrepublic.com/article/10-questions-software-developers-should-expect-in-a-job-interview/>
10. Texto bem longo e muito bom sobre contratação de programadores. Adoro as coisas que o Joel escreve. <https://www.joelonsoftware.com/2006/10/25/the-guerrilla-guide-to-interviewing-version-30/>
11. Áreas de programação: <https://www.quora.com/What-are-the-different-areas-of-programming>
12. API: <https://www.howtogeek.com/343877/what-is-an-api/>
13. Git Branches: <https://nvie.com/posts/a-successful-git-branching-model/>
14. Terminal: https://itconnect.uw.edu/learn/workshops/online-tutorials/web-publishing/what-is-a-terminal/
15. Terminal: https://en.wikipedia.org/wiki/Command-line\_interface
16. SSH: https://www.ssh.com/ssh/protocol/
17. SCP: <http://man7.org/linux/man-pages/man1/scp.1.html>
18. Architecture: <https://www.quora.com/What-is-an-architecture-diagram>

**Perguntas Respondidas:**

1. Por que uma empresa contrata um programador? O que é esperado que ele faça?
2. Que problema o Git resolve? Como?
3. O que é o GitHub?
4. Qual a diferença entre Git e GitHub?
5. Qual o ciclo de desenvolvimento de um software? O que é um deploy?



1. O que é metodologia de desenvolvimento de software? Quais são as mais famosas?
2. O que é QA? Pra que serve?
3. O que significa fazer o fork de um repositório no GitHub?
4. O que significa fazer o clone de repositório?
5. O que é um commit?
6. O que significa fazer um push?
7. O que significa fazer um pull?
8. O que é uma branch?
9. O que significa fazer um merge?
10. O que é um pull request no GitHub?