Atividade sobre Git e Github

**1. Descreva a biografia completa de quem criou o Git e o GitHub e explique qual a diferença entre eles.**

* Quem criou o GitHub?

Chris Wanstrath, J. Hyett, Tom Preston-Werner e Scott Chacon. A empresa criadora do GitHub foi a GitHub Inc., criada em 2007 em São Francisco nos EUA.

* Quem criou o Git?

O git é um projeto de código aberto maduro e com manutenção ativa desenvolvido em 2005 por Linus Torvalds. Linus Torvalds - Engenheiro de Software, formado em Ciência da Computação, nascido na Finlândia, desenvolvedor do núcleo Linux, utilizado nos SOs Linux, Android e Chrome.

* Diferença entre Git e GitHub

O Git é um software de controle de versão, já o GitHub é como se fosse uma rede social de pessoas desenvolvedoras, ou seja, uma plataforma na qual podem compartilhar projetos.

**2. Para que serve?**

O Git serve para registrar quaisquer alterações realizadas no código, funciona como um sistema de versionamento de códigos, possibilitando que o programador possa voltar a versões anteriores do código. O GitHub serve como uma rede social de programadores, na qual é possível interagir, seguir, visualizar código de outros programadores e compartilhar os próprios códigos. O Git e o GitHub são utilizados por programadores do mundo todo. Essas ferramentas são responsáveis por possibilitar um gerenciamento contínuo e pontual das versões de códigos de um determinado projeto.

**3. Qual o principal objetivo da plataforma?**

Git

O Git é um software de controle de versão. Com ele, conseguimos gerar registros do nosso projeto mantendo o controle em quais arquivos foram alterados, quando foram alterados e por quem. Assim, é possível manter uma linha do tempo de um projeto e voltar nesses registros quando for necessário. Por exemplo, algo do projeto parou de funcionar, mas antes estava funcionando. Outra vantagem de utilizar o Git é a possibilidade de termos múltiplos desenvolvedores trabalhando no mesmo projeto, facilitando o “cada um faz a sua parte e depois juntamos”. Essa característica do Git somada a uma ótima comunicação do time de devs, possibilita projetos grandes serem concluídos em intervalos de tempo cada vez menores.

Github

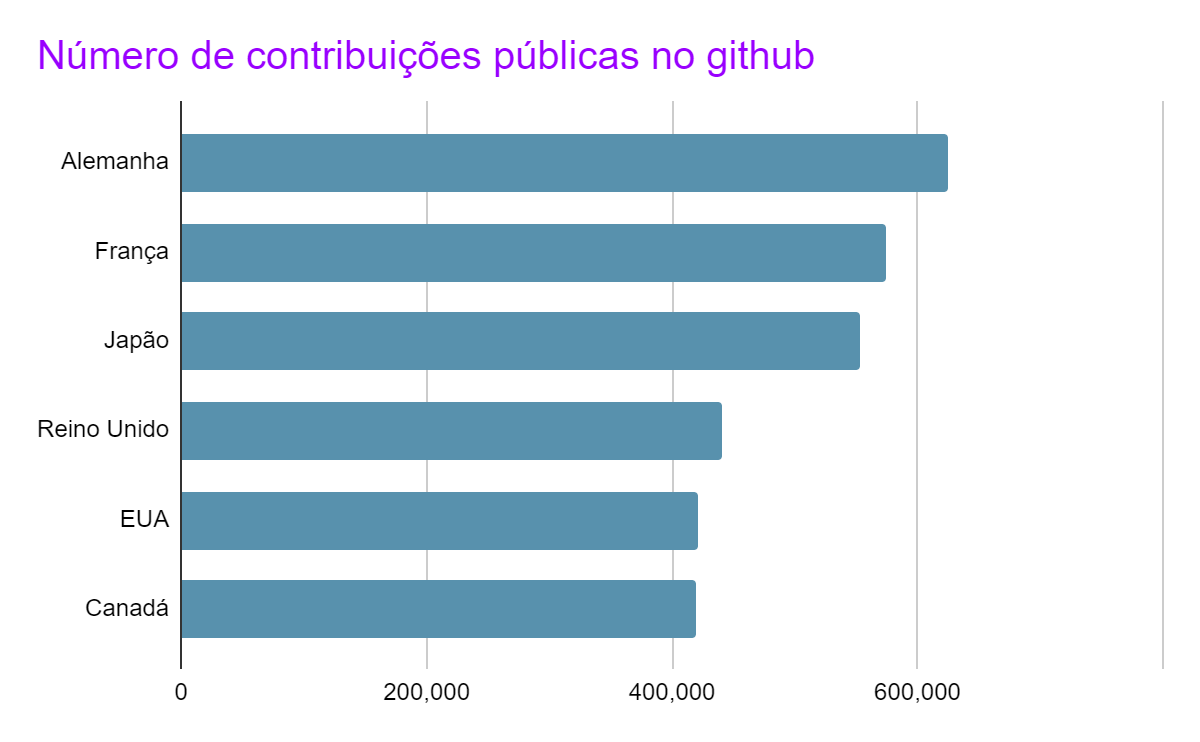
O GitHub é uma espécie de rede social de desenvolvedores. Enquanto o Git é um software, o GitHub é uma plataforma onde você pode subir seus projetos, criar seu portfólio e compartilhá-lo com o mundo. O GitHub funciona usando o Git “por de baixo dos panos” para que devs possam desenvolver e atualizar os seus projetos sozinhos ou de forma colaborativa com outros devs.

**4.** **Quantos usuários e empresas utilizam no mundo? Quais países que mais utilizam? (classificação, números e gráficos)**

A plataforma Github conta com mais de 83 milhões de desenvolvedores e 4 milhões de empresas.

Países que mais utilizam

* Alemanha: 624,543
* França: 575,165
* Japão: 552,879
* Reino Unido: 441,055
* EUA: 421,453
* Canadá: 418,958



5. Quais os principais comandos utilizados e quais os termos utilizados no GitHub?

* O Git init cria um repositório vazio.
* O Git clone duplica o repositório.
* O Git add adiciona novos arquivos ou modifica existentes no index, antes de fazer o commit.
* O Git status checa se os arquivos estão no repositório.
* O Git diff mostra diferenças entre uma árvore e o index, entre árvores, entre blobs ou entre arquivos
* O Git branch é uma ramificação/bifurcação realizada no projeto principal, ou seja, é a criação de um novo caminho/branch para o código visando não alterar o caminho/branch principal.
* O Git merge serve para unificar ramos de códigos de desenvolvimentos distintos criados pelo Git branch.
* O Git push é usado para enviar o conteúdo de um repositório local para um repositório remoto.
* O Git pull é utilizado para buscar e baixar conteúdo de repositórios remotos e fazer a atualização imediata no repositório local para que os conteúdos sejam iguais e atualizados.
* O Git fetch é o oposto do git push, visto que esse comando é responsável por realizar o download dos commits, arquivos e referências de um repositório remoto para um repositório local.
* O Git remote possibilita criar, visualizar e deletar conexões com outros repositórios, listando os nomes de cada repositório remoto manipulado especificado pelo usuário.
* O Git commit é o comando utilizado para registrar as alterações realizadas nos arquivos de um determinado repositório.
* O Git show é utilizado para exibir informações acerca de objetos do git, como blobs, trees, marcações e commits.
* O Git log permite visualizar as alterações realizadas em um determinado repositório.
* O Git config é a função que possibilita definir valores de configuração para o git em nível global ou local e, ao executá-lo, o arquivo de texto de configurações localizado no computador local é alterado.