

GITHUB

 lição por Haydee Svab

Sabe aquele trabalho em grupo ou mesmo projeto individual que, na sua produção, conta com um monte de arquivos e versões?

Embora possamos organizar tudo em pastas nas nossas máquinas ainda assim demanda um esforço ter um controle de versão de cada arquivo. Esse cenário fica mais confuso se forem muitas pessoas colaborando ao mesmo tempo em diversos arquivos, não é?

Aí, neste momento, você deve ter pensado: “para isso uso o GoogleDocs” ou “aí a gente põe tudo no Drive”. Mas sabia que existe uma ferramenta que possibilita:

- guardar vários arquivos;
- acessar esses arquivos a partir do navegador;
- controlar a versão de cada documento;
- guardar o histórico de cada versão;
- abrir um projeto em vários outros, com controle de versão e histórico, sem perder a origem?

Apresentamos então o GitHub!

O que é o Git?

É um repositório, ou seja, um sistema de controle de versões distribuído – todas pessoas que trabalham num projeto terão cópia de todo o histórico do projeto.

O que é o GitHub?

É um site em que você pode carregar a sua versão do repositório Git.
<https://github.com/>

Você pode entrar no **GitHub** acessando <https://github.com/>. Lá você pode criar uma conta clicando em **Sign Up**.

Depois de criar login e senha, você acessa o seu repositório clicando em **Sign In**. Sempre que um novo projeto **Git** for feito, um **branch master** será criado automaticamente. Essa é a espinha dorsal do seu projeto, aqui ele nasce, e aqui devem ser consolidados os produtos que estarão no resultado final. Só que do primeiro passo até o resultado final, muita experimentação acontece, erros, acertos, e até mesmo nascimento de projetos novos que depois criam pernas próprias.

Para experimentar, como numa folha de papel à parte, a partir da ideia original sem alterá-la é comum fazermos um **branch de feature**.

Mas afinal, o que é um **branch**? **Branch** (ou ramo), é uma derivação do projeto raiz que congrega uma série de **commits**.

Vamos lá, imagine uma proposta comercial, em que foram pontuados apenas os títulos e algumas palavras-chaves para seu conteúdo. Mas aí você tem cenários diferentes: clientes mais fieis, clientes que não pagam em dia e novos clientes. A abordagem de sua proposta comercial pode mudar bastante dependendo do cliente em foco, certo?

É possível ter uma proposta esboço entrando como **branch master**, e a partir daí, pensar em um tipo de proposta diferente para cada tipo de cliente (serão **branches de features** diferentes). Mas os vários profissionais têm diferentes ideias para cada um desses clientes e cada um deles pode colaborar com essa proposta, fazendo **commits**. Assim, cada vez que uma alteração for feita, será feito um **commit** onde se estiver trabalhando.

Sempre que fizer um **commit**, ajuda a organizar o processo todo, escrever muito sucintamente o motivo daquele commit (que pode ser de acréscimo, decréscimo ou alteração do existente) – isso a gente chama de mensagem de **commit**. Isso é extremamente útil para depois a entender e resgatar histórico das tomadas de decisão nos projetos.

E sabe aquela lista de coisas a fazer, mas que não dá para fazer tudo de uma vez? Tem gente que cola **post-its**, gente que faz listas num caderno, aqui, no **GitHub**, cada uma dessas tarefas vira uma **issue**.

Ok, um monte de sugestões entraram via commits e estão organizadas em diversos branches, mas agora, se você quiser consolidar algumas dessas ideias você precisará dar um merge daquele **branch de feature** que gostou com o **branch master**.

Tomamos por exemplo o projeto Radar Parlamentar, disponível em <https://github.com/radar-parlamentar/radar>, para você entender melhor a funcionalidades do repositório.

Visite e contribua para aumentarmos o nosso número de colaboradores!