

## Entrevista 21 - Entrevistado

Transcribed by [TurboScribe.ai](#). [Go Unlimited](#) to remove this watermark.

Eu sou um front-end senior. O meu dia-a-dia é desenvolver novas features para o cliente, também resolução de bugs, ajudar no deployment das novas features. Basicamente isso.

Atualmente, um projeto apenas. Incluindo eu, são seis desenvolvedores. Desculpa, não percebi? Sim, exclusivamente desenvolvedores.

Não entendi, a parte do código? Ah, claro. Então acho que é bom eu explicar mais ou menos como é estruturado. A gente usa o modelo de feature.

Então temos a nossa main, temos a branch de dev baseada na main. E a partir da dev, nós fazemos as nossas features baseadas naquele pedido, naquele requisito que o PO escreveu. Depois que terminamos todo o desenvolvimento desta feature, criamos um mass request dessa feature branch para a branch de dev.

Então, a partir daí, pedimos para os outros developers validarem o nosso código, identificarem qualquer erro, então sugestões de códigos ou até melhorias mesmo de qualquer gênero. E quando pelo menos um developer aprova o mass request, nós seguimos com o mass, ela passa nas pipelines. E quando passa nas pipelines e esse código é feito o mass para a branch de dev, aguardamos até o final da semana, onde fazemos o nosso deploy para a stage.

E no final da sprint, quando o PO validou todas as nossas features feitas durante a sprint, nós agendamos um dia, geralmente na terça ou quarta-feira, para fazer o deployment para a production. Boa pergunta. Não, se mudava qualquer coisa era muito sutil, mas era basicamente isso para todos os outros projetos que eu já passei.

Ok, no início é um pouco difícil de entender, mas a partir do momento que você faz uma ou duas vezes de cada parte desse esquema de vacinamento, fica muito mais fácil de entender o que está acontecendo e de como resolver um problema, por exemplo, quando é necessário fazer uma hotfix. Você já sabe de onde basear sua branch e como lidar a partir disso, para que tudo esteja na mesma página. Como eu disse, logo no início é um pouco difícil de entender, então para os developers mais novos pode tomar um tempo e acaba dando abertura para novos erros serem cometidos.

Então, na minha opinião, é mesmo esse ponto de entrada que acaba sendo mais um downside desse formato. É assim, logo no ponto de entrada, como eu disse, depois que a pessoa passa por esse processo uma ou duas vezes já fica bem mais fácil de entender. No ponto de vista do novo developer, olha, eu acho que ajuda bastante.

Como eu disse, a parte mais difícil é só no início, então digamos que na segunda sprint esse novo developer já sabe mais ou menos como encaminhar e também, pelas equipes

que passei, por exemplo, todas eram muito prestativos, então a qualquer momento esse developer mais recente pode pedir ajuda ao restante da equipe, caso não esteja à vontade ainda. Então eu acho que é um bom modelo, sim, para novos developers. Não, pode repetir a pergunta, desculpa.

Certo, certo. Ah, entendi. Não, já é uma empresa estabelecida, presente em vários países.

Não, geralmente evitamos vários developers terem branches que se relacionam feature-wise justamente para prevenir uma grande quantidade de conflitos. Então isso geralmente só acontece mesmo quando alguém altera alguma coisa muito genérica que afeta outros componentes, ou então quando é inevitável e dois developers têm que trabalhar no mesmo tema, ou então no mesmo módulo da aplicação.

**Transcribed by [TurboScribe.ai](#). [Go Unlimited](#) to remove this watermark.**