Palestrante 2  
Beleza, tudo bem se eu gravar essa reunião para depois eu usar a transcrição dela para analisar os dados?

Palestrante 1  
Não tem problema.

Palestrante 2  
Perfeito. Ótimo, então eu vou colocar pra gravar aqui. Então, <entrevistado>, a ideia é a seguinte: a entrevista é dividida em 3 partes.  
Primeiro, uma parte bem rápida, só para caracterizar o entrevistado. Eu vou refazer aquelas perguntas que já te fiz sobre a sua experiência e o seu papel hoje, sua posição na empresa.  
Depois, duas partes sobre o bot: uma delas é sobre se você acha que o bot poderia ajudar a gerenciar a dívida técnica, e a segunda parte é sobre o quão fácil seria utilizá-lo. Pode ser o mais sincero possível — quanto mais sincero, melhor.

Palestrante 2  
Legal. Então, a sua atual posição na empresa, qual que é?

Palestrante 1  
Eu sou Senior Mobile Engineer, né? Ou Senior Software Engineer. Porque não necessariamente atuo somente na frente de mobile. Às vezes atuo também, mas não é minha expertise, no back-end.

Palestrante 2  
Ah, entendi.

Palestrante 1  
Mas a minha pesquisa lá na <empresa> é especificamente em Flutter, não é? Então sou Senior Mobile Engineer.

Palestrante 2  
Legal, legal. E quantos anos de experiência você tem ao todo? E quantos anos nessa posição?

Palestrante 1  
Sou desenvolvedor de software desde 2012. Somando com estágios, dá uns 13 anos em desenvolvimento. Mais ou menos 12 ou 13 anos. Comecei estagiando em 2012, mas oficialmente dá pra quantificar a partir de 2013.  
Entrei na universidade em 2010 e já comecei a estagiar. Peguei alguns projetos voltados a mobile, naquela época nativo, com Java. Atualmente trabalho com Flutter desde o release, em dezembro de 2018, e profissionalmente desde março de 2019.

Palestrante 1  
Quando vim para Maringá e para a <empresa>, faz 4 anos, estritamente com Flutter.

Palestrante 2  
Legal, legal. Beleza, a outra pergunta seria sobre as suas tarefas no dia a dia, mas você já deu uma pincelada nisso. A gente pode passar para a parte de avaliação do bot mesmo.  
Como eu disse, vamos dividir em 2 partes: uma sobre o quão útil você acha que o bot pode ser e outra sobre o quão fácil é utilizá-lo.  
No geral, como foi a experiência com o bot? Tem algum comentário geral em relação à ferramenta? O que você achou?

Palestrante 1  
Eu achei bem simples. A parte de instalação é bem tranquila, parecida com qualquer app de instalação do GitHub. A gente vai no marketplace, aceita os termos, coloca os movimentos externos, escolhe o repositório ou se vai ser em todos. Bem simples.  
O bot tem um config default, o que viabiliza bastante, porque a configuração foi praticamente zero. Não precisei fazer nada, isso foi muito bom.

Palestrante 1  
É o que a gente chama de “plug and play”, né? Você instala e já roda. Achei bem fácil mesmo.

Palestrante 1  
Em relação à classificação, achei muito interessante. Um dos fatores primordiais do desenvolvimento de software é entender a questão da dívida técnica. Acho que é um potencial muito grande de uso, principalmente em projetos grandes.  
Em pequenos projetos talvez não seja tão necessário, mas em grandes ajuda demais. Garante que todas as issues sejam analisadas com o mesmo critério: débito técnico ou não. Isso é muito interessante.  
A integração é fácil. Recebi aqui nos meus e-mails duas notificações. Foi bem rápido, assim que a issue foi gerada. Bem interessante, ajuda muito, principalmente para equipes mais sêniores.

Palestrante 2  
E você acredita que, como mencionou a questão de classificar as issues de forma padronizada, isso ajudaria você a identificar mais facilmente o que é dívida técnica e o que não é?

Palestrante 1  
Com certeza. Imagine um projeto grande e público no GitHub, recebendo dezenas de issues de pessoas diferentes. Muitas vezes não conseguimos classificar se aquilo é uma issue comum ou um débito técnico.  
O débito técnico precisa ser resolvido o mais breve possível. Então seria uma ferramenta muito útil, algo que a gente poderia analisar e colocar mais à frente em nossos projetos. Gostei muito, ajuda demais.

Palestrante 1  
Outro ponto interessante: com o bot e a inteligência artificial, conseguimos reduzir o trade-off do PM ter que classificar o que é prioritário ou não. Com essa classificação fica mais fácil conversar com o PM/PO e mostrar que aquilo é de fato prioritário. Isso é muito interessante.

Palestrante 2  
E você acha que essa questão do bot fazer classificações também ajudaria a monitorar as issues ao longo do tempo?

Palestrante 1  
Ah, com certeza. A partir das notificações já é essencial. Imagine depois integrar com o Telegram, que é até mais fácil que com o WhatsApp. Seria fantástico, principalmente em projetos com muitos usuários e desenvolvedores.  
Exemplo: abriram uma issue sobre login inseguro, sem criptografia. Se o bot identificar isso como débito técnico e gerar notificação, ajuda muito.

Palestrante 2  
Acho que na parte de utilidade era isso, <entrevistado>. Agora vamos para a parte de facilidade de uso. Você já mencionou que foi fácil instalar e configurar o bot. E para aprender a operá-lo, usar os comandos?

Palestrante 1  
Bem simples. Pela documentação, foram 3 ou 4 comandos. Tem um essencial, o bot help, que já mostra os comandos de ajuda. Muito tranquilo.

Palestrante 2  
Legal. E em relação à documentação? Você achou clara e organizada? Foi tranquilo usar a documentação para entender como o bot funciona?

Palestrante 1  
Sim, bem simples e direta. Passei uns 10 minutos conversando, mandei o link e já começamos a usar. Documentação bem objetiva.

Palestrante 2  
E depois que você leu a documentação e mexeu com o bot, você acha que seria tranquilo lembrar dos comandos caso não tivesse acesso à documentação?

Palestrante 1  
Sim, bem simples. Os comandos são diretos: tdbot playbook, tdbot help, tdbot label. Muito bom mesmo, tranquilo.  
Obviamente, se no futuro houver novos comandos, aí seria essencial ter a documentação por perto. Mas no estado atual, os comandos são básicos e fáceis de memorizar.

Palestrante 2  
E por fim, <entrevistado>, a última pergunta: você sentiu falta de alguma funcionalidade ou pensou em alguma que poderia ser implementada no bot?

Palestrante 1  
Não, bem tranquilo. Talvez no futuro, com atualizações, o bot possa identificar automaticamente uma issue aberta, sem precisar de comando manual. Mas isso é algo para frente, não atrapalha o uso. Bem bacana.

Palestrante 2  
Legal, <entrevistado>. Acho que era basicamente isso. Vou encerrar a gravação aqui.