Pesquisador. Muito obrigado mais uma vez por participar dessa entrevista. Ela tem como objetivo compreender os processos de criação e de manutenção de código de testes e como é realizada a verificação da qualidade desse código de teste de unidade. Não há respostas certas ou erradas ou desejáveis ​​ou indesejáveis. Dessa forma, senta-se à vontade para dizer o que realmente pensa. Reforçando um item do termo de consentimento, nossa conversa está sendo gravada. O objetivo disso é para que eu possa obter todos os detalhes, mas ao mesmo tempo ser capaz de manter uma conversa atenta com você. Garanto que todos os seus comentários permanecerão confidenciais e que no relatório final conterá os comentários dos participantes sem nenhuma referência aos indivíduos. Tudo bem?

E. Tudo bem, muito obrigado.

P. Como você começou a trabalhar com teste de software?

E. É parte do trabalho de desenvolvimento do cargo que ocupo. A gente trabalha em projetos diversos e tanto testar quanto desenvolver o software é parte do trabalho. Então é esperado de nós que a gente teste o código que a gente desenvolveu.

P. Quais foram suas fontes de aprendizado sobre código de teste?

E. Eu tive cadeiras relacionadas a isso durante a faculdade e antes de entrar no trabalho que estou agora passei por um programa de treinamento da Apple que teve instrutores que nos ensinaram a testar código dentro do framework da Apple.

P. Em quais linguagens você cria testes?

E. Em Swift Care é a linguagem que eu mais trabalho e que é usado como padrão na área que eu trabalho na empresa que é desenvolvimento IOS.

P. Qual plataforma do seu projeto atual?

E. IOS.

P. Como é seu processo de criação de teste de unidade?

E. O teste de unidade geralmente a gente faz a partir da tarefa que a gente tem criada que depende da arquitetura do projeto. O projeto que a gente está atualmente trabalha com MVVM (Model-view-viewmodel), então a gente foca os testes unitários em View model que a gente desenvolveu. Então dentro de cada história a gente tem as tarefas para desenvolver as View Models e encaixar elas nas View Controlers e no final de cada tarefa que envolva criar ou atualizar ou mexer na View Model, a gente tem uma tarefa associada que é a criação de teste unitário para ela. Então a gente analisa tudo que foi criada para essa view model foi feita de acordo e depois é revisado por outros desenvolvedores também.

P. Existe algum fluxograma, template na empresa que padronize esse processo de criação de teste de unidade?

E. Não. Que eu saiba não. Do teste de unidade que a gente desenvolve também varia de acordo com o cliente. Então pode ter exigências do cliente a respeito, mas do lado da empresa não existe nenhum fluxograma ou requisito.

P. Quais ferramentas são utilizadas para criar/ executar teste?

E. O próprio Xcode, que é a ferramenta que a gente usa para desenvolver. Ele tem o framework de testes que é o Xcode test case, se não me engano e é esse que a gente utiliza mesmo.

P. Como é o processo de verificação de qualidade do código de teste criado?

E. A verificação da qualidade é feita pela opinião dos outros desenvolvedores, juntamente com com o líder do projeto, o olheiro. Scrum master seria o nome mais apropriado talvez. Ele gerencia as atividades, o que tem para fazer e qual a prioridade de cada tarefa. Eles analisam todos os códigos que são integrados e além disso, a gente preza pela cobertura de código, apesar de saber que isso não é um indicativo final de qualidade.

P. Quantos desenvolvedores fazem esse código de teste?

E. Varia de acordo com o cliente. No projeto que estou atualmente a gente precisa de no mínimo 3 outras pessoas revisando, sendo que uma delas é do lado do cliente. Pode ser o líder de projeto do cliente quanto um outro desenvolvedor deles.

P. Passando para o processo de manutenção do código de teste, como ele é realizado?

E. Isso varia de acordo com as tarefas. Tem vezes que o controle sobre isso não é muito grande. A gente pode ter alterações em View models que impacte nos testes, sem que seja percebido. Então a gente executa os testes sempre no final das sprints, mas por vezes durante as sprints a gente comete erros que acabam prejudicando os testes que tem que ser corrigidos mais à frente.

P. O que você conhece sobre test smell?

E. Smells seria sobre cheiro ruim, né? A expressão?

P. Isso.

E. eu conheço um pouco sobre teoria que tive sobre isso na faculdade se não me engano.

P. Voltado para o código do software ou para o código de teste?

E. Voltado para o código de software.

P. O conceito para o teste ele é o mesmo. Tanto que test smell descende de code smell que é o que você viu na graduação. Só formalizando o conceito e tirando alguma dúvida caso você tenha, “test smells são pontos fracos na estrutura do código de teste, tendo um impacto negativo na sustentabilidade da atividade, pois os tornam complexos, difíceis de entender e modificar, prejudicando a repetibilidade, independência e estabilidade dos testes.”

E. Certo.

P. Você possui alguma dúvida sobre esse tema?

E. Acho não.

P. Eu vou passar para as perguntas seguintes, mas se você tiver alguma dúvida é só perguntar.

E. Tudo bem.

P. Como test smells são tratados dentro do processo de criação do código de teste?

E. Pelo conceito que me destes, a gente trabalha de acordo com o que o cliente prefere quanto as tasks. No projeto atual é o líder técnico do cliente que define as diretrizes dos testes então eles têm um jeito de testar que eles já implementaram e a gente segue esse padrão. Que diz mais respeito a "instanciação" do que vai ser testado e do jeito que tu testa mesmo.

P. Ele estaria muito mais relacionado a cobertura e “instanciamento” do que a verificação de smells, é isso?

E. Isso.

P. Isso ocorre tanto na criação quanto na manutenção ou na manutenção ocorre de alguma forma diferente?

E. A manutenção ocorre do mesmo jeito. Envolve os dois.

P. Para você, como seria possível prevenir/evitar a inserção de test smell no código de teste no momento da criação do código de teste? Quais técnicas de programação, processos organizacionais podem ser adotados?

E. Tornando de uma forma mais clara o que você vai testar do jeito mais separado possível. Dividindo em mais casos de teste talvez.

P. Estamos chegando no final da entrevista. Você tem alguma pergunta para mim, quer acrescentar alguma informação ou ainda tem alguma sugestão para melhorar nossa entrevista?

E. Não. Eu acho que foi bem sucinto e direto ao ponto.

P. Muito obrigado por ter participado da entrevista. Eu vou finalizar a gravação.

E. Eu que agradeço a participação.