**Questionário DyeVC**

1. Com que frequência você utilizou o DyeVC?
2. Com que frequência você manteve a tela principal do DyeVC aberta?
3. Você normalmente trabalhou fisicamente próximo ao seu colega de trabalho?
4. Você teve ciência do que seus colegas estavam fazendo através das notificações do DyeVC?
5. As notificações apresentadas na barra de tarefas foram perturbadoras?
6. Quando você notou que havia commits com alguém que você ainda não possuía, você:
   1. Conversou com seu colega
   2. Ignorou a informação e continuou trabalhando
   3. Realizou um pull imediatamente
   4. Terminou de fazer as alterações que estava fazendo e realizou um pull
7. Com que frequência você efetuou commit de seu código?
8. Com que frequência você precisou efetuar merge de código que já havia sido comitado?
9. Quando você não estava fisicamente próximo de seus colegas, o DyeVC foi mais útil do que quando próximo? Se sim, por que?
10. Qual a funcionalidade mais útil do DyeVC e por que?
11. Que melhorias você sugere?
12. Classifique as seguintes afirmações em uma escala de 1 (= concordo fortemente) a 7 (= discordo fortemente)
    1. Foi fácil para mim saber se meus colegas estavam efetuando commits no projeto
    2. Não foi útil ver os commits que meus colegas estavam realizando
    3. Eu gostei de saber que meus colegas estavam realizando commits, ainda que eu não me preocupasse imediatamente em verificar conflitos
    4. Eu não me senti confortável com outras pessoas sabendo quando estou realizando commits
13. Classifique as seguintes melhorias em termos de em uma escala de 1 (= concordo fortemente) a 7 (= discordo fortemente)
    1. Criar um filtro de commits, que permita escolher os arquivos ou os desenvolvedores de interesse
    2. Permitir ao desenvolvedor comparar a versão mais atual de um arquivo com a sua versão
    3. Mostrar a situação (online / off-line) de cada repositório
    4. Incluir o nome das tags na história de commits
    5. Prover um mecanismo para realizar pull / push / merge automaticamente

Demographic

Q1. What is the type of project that you spend most of your time on?

Q2. What is the extent of your programming experience?

Q3. How old is the project that you typically work on?

Q4. What type of VCS paradigm do you prefer?

Q5. What is the VCS tool that you use most often?

Q6. What is the size of your software development team?

Commit practice

Q7. How often do you commit your changes?

Q8. If you chose “It depends”, what does it depend upon? \*

Q9. When you commit, how do you group your changes?

Q10. If applicable, what criteria do you use to split your commits? \*

Q11. If applicable, do you squash your commits?

Q12. If you squash your commits, what are your reasons for squashing? \*

Q13. Which of the following VCS do you ﬁnd the most natural to commit changes?

Q14. Why do you ﬁnd it most natural? \*

Issue tracking

Q15. Does the project where you sped most of your time have a commit policy?

Q16. Do you use an issue tracking system?

Q17. If yes, do you work on more than one issue at a time?

Q18. If yes, are the issues related?

Q19. If you work on several issues at once, do you commit each issue separately?

Q20. How often do you share changes with other members?