**TRABALHO ENGENHARIA DE SOFTWARE II – PROFº CARLOS ROBERTO PEREIRA PADOVANI**

**ALUNOS:** Amanda de Oliveira Ribeiro

Hugo Tieghi

Pamela Roustini da Fonseca

Sergio Di Fiore

**TEMA: PROGRAMAÇÃO EM PAR**

**Definição**

A programação em pares consiste em dois programadores que compartilham uma única estação de trabalho (uma tela, teclado e mouse para o par). O programador no teclado é geralmente chamado de “driver”, o outro, também ativamente envolvido na tarefa de programação, mas focando mais na direção geral é o “navegador”; espera-se que os programadores troquem de funções a cada poucos minutos ou assim.

**História**

Fontes verificáveis ​​sugerem que a programação em pares, em sua forma moderna, existe desde muito antes do movimento Ágil:

* 1992: “Dynamic Duo” é o termo cunhado por Larry Constantine, relatando uma visita a Whitesmiths Inc., um fornecedor de compiladores fundado por PJ Plauger, um dos implementadores de C: “Em cada terminal havia dois programadores! Claro, apenas um programador estava realmente cortando o código em cada teclado, mas os outros estavam espiando por cima dos ombros. ” Whitesmiths existiu de 1978 a 1988.
* 1993: “Os benefícios da colaboração para alunos programadores” por Wilson et al. é um dos primeiros estudos empíricos que indicam os benefícios do emparelhamento para tarefas de programação especificamente. Os estudos posteriores são mais abundantes e impulsionados pelo desejo de “validar” a programação em pares depois que ela já ganhou popularidade por meio da [Programação Extrema](https://www.agilealliance.org/glossary/xp/) .
* 1995: o padrão “Developing in Pairs” recebe uma breve descrição no capítulo de Jim Coplien “A Generative Development-Process Pattern Language” do primeiro livro de padrões, “Pattern Languages ​​of Program Design”.
* 1998: em “Chrysler vai aos extremos”, o primeiro artigo sobre Programação Extrema, a programação em pares é apresentada como uma das principais práticas da equipe C3; é mais tarde descrito formalmente como uma das “doze práticas” originais do XP
* 2000: (ou anterior) - as funções do motorista e do navegador são apresentadas para ajudar a explicar a programação em pares; a referência mais antiga conhecida é uma [postagem em](http://tech.groups.yahoo.com/group/extremeprogramming/message/12405) uma [lista de discussão](http://tech.groups.yahoo.com/group/extremeprogramming/message/12405) ; observe, no entanto, que a realidade dessas funções tem sido contestada, por exemplo, o artigo de Sallyann Bryant " [Programação em pares e o misterioso papel do navegador](http://www.sciencedirect.com/science/article/pii/S1071581907000456) "
* 2002: “ [Pair Programming Illuminated](http://www.amazon.com/dp/0201745763) “, de Laurie Williams e Robert Kessler, é o primeiro livro dedicado exclusivamente à prática e discute sua teoria, prática e os vários estudos até aquela data
* 2003: um artigo anônimo no C2 Wiki descreve [a Programação de Ping-Pong](http://www.c2.com/cgi/wiki?PairProgrammingPingPongPattern) , uma variante moderadamente popular que combina o emparelhamento com o desenvolvimento orientado a testes.
* 2015: James Coplien publica [Two Heads are Better Than One,](https://computingnow.computer.org/web/agile-careers/content?g=8504655&type=article&urlTitle=two-heads-are-better-than-one) que fornece uma visão geral da história da programação em pares que remonta a meados dos anos 1980, se não antes.

**Pontos para ficar atentos**

Ambos os programadores devem sempre estar atentos e envolvidos com a tarefa durante o período em que estiverem trabalhando, caso isso não ocorra não poderá se tirar vantagem dessa metodologia;

Uma questão oposicionista em questão ao uso dessa metodologia são os custos, dizendo que irá dobrar “seus gastos”. Sendo que na verdade poderá ser bem vantajoso considerando o prazo de entrega e futuras correções que poderiam acarretar futuramente próximo da entrega do projeto ao cliente.

Deve-se ter uma interação entre os 2 programadores que estiverem trabalhando junto, para que consigam se manterem atentos a possíveis deslizes que possa ocorrer com qualquer uma das partes do Par.

Os pares devem manter-se unidos e procurar entrar em acordo para que nada atrapalhe o desenvolvimento do trabalho em curso. Se algo incomoda no parceiro de trabalho procurar resolver essas diferenças antes de cada trabalho.

O ambiente de trabalho deve estar devidamente disposto e convidativo para um bom desenvolvimento do trabalho. O nível de ruído na sala deve ser controlado, a conversa entre os pares deve ser em um nível agradável e que respeite a todos os outros que estão no mesmo ambiente.

**BENEFICIOS ESPERADOS**

Melhor qualidade no código, uma conversa aberta entre os pares deixa o nível de atenção das partes mais alto. Deixa o nível de complexidade mais claro e também os detalhes ocultos melhor entendidos, fazendo com que os riscos de erros ou becos sem saída se reduzam.

Melhora o nível de conhecimento entre a equipe, onde um desenvolvedor que não tenha muito conhecimento no assunto esteja trabalhando com um desenvolvedor que já tenha mais experiência na área.

Melhor as técnicas e as habilidades dos programadores juniores a medida que aprendem micro técnicas com os membros da equipe mais experientes.

A Função de Coordenação fica mais leve pois há pares a serem coordenados ao invés de indivíduos solitários, com isso é melhor compreendido as tarefas passadas para o par.

Os pares são mais abertos a interrupções em comparação ao individual, pois quando o par for atender a um chamado um membro do par atende ao chamado enquanto o outro continua executando a tarefa e após a volta do membro pode ser retomada a tarefa sem ter perdido o foco.