

## Questionário de adesão – Programação Extrema (XP)

Equipe/Projeto: \_\_\_\_\_

Data: \_\_/\_\_/\_\_

**A) Tempo de educação formal em Informática ou Ciência da Computação (anos)**

(Para responder esta pergunta, considere a duração normal do curso e não o tempo que você demorou para concluí-lo, por exemplo, se você fez graduação em Computação e já concluiu o primeiro ano do mestrado, responda 5 anos (4 anos da graduação mais 1 de mestrado, mesmo que você tenha demorado 6 anos para concluir a graduação)

0  1  2  3  4  5  6 ou mais

**B) Tempo de experiência profissional em Informática (anos)**

0  1  2  3  4  5  6  7  8  9  10 ou mais

**C) Número de cursos específicos em Programação Orientada a Objetos**

(por exemplo: MAC 110 em Java, POO, Tópicos de POO, SOD, Padrões de Projeto, etc.)

0  1  2  3  4 ou mais

**D) Responda às perguntas abaixo dando uma nota de 0 a 10 para cada uma das práticas de Programação Extrema (XP), considerando o nível atual da equipe (Atual) e o nível que considera desejável (Ideal):**

Prática	Comentários
<p><b>Programação Pareada</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Dois pessoas trabalham juntas no mesmo computador. Elas trocam de papel constantemente atuando como digitador ou revisor, pensando no trabalho que estão realizando.</i></p> <p>10 Nós não gostaríamos de escrever qualquer código crítico sem pairar. Nós fazemos o rodízio constante de pares.</p> <p>8 Nós geralmente trabalhamos em pares.</p> <p>6 Nós geralmente fazemos discussões no quadro branco, conversamos no chat ou em visitas ao laboratório. Algumas pessoas programam em pares no teclado, mas alguns preferem não tentar</p> <p>4 Nós tentamos pairar mas não conseguimos devido a desencontro de horários e reuniões. Algumas pessoas são muito rápidas ou devagares para que eu tenha paciência de sentar junto. De qualquer forma, nossos móveis tornam o pareamento difícil.</p> <p>2 Eu me distraio muito quando alguém me interrompe. Meu companheiro de trabalho me pede para não ter tantos visitantes.</p> <p>0 Eu uso fones de ouvido para que as pessoas não me interrompam. Na verdade, eu prefiro trabalhar em casa com o telefone fora do gancho e com o programa de chat configurado no modo "Ocupado".</p>
<p><b>Releases Pequenas</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Nós entregamos iterações pequenas e mais frequentes ao nosso cliente.</i></p> <p>10 A cada 1 ou 2 semanas nosso cliente pode ter uma nova versão funcionando do nosso sistema.</p> <p>8 Nós temos iterações mensais. O cliente pode escolher novas funcionalidades para a próxima versão.</p> <p>6 Dentro de alguns meses nós disponibilizamos uma nova iteração para nosso cliente.</p> <p>4 Nós entregamos versões beta e patches de correção umas 4 vezes ao ano, com entregas maiores em ciclos de 8 a 12 meses.</p> <p>0 Temos visão a longo prazo. A versão 1.0 do ano que vem te deixará entretido até que as verdadeiras funcionalidades sejam implementadas na versão 2.0 dentro de 18+ meses.</p>
<p><b>Integração Contínua</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Quando trabalhando em uma funcionalidade, sincronizo e disponibilizo código:</i></p> <p>10 Diversas vezes por dia</p> <p>8 Uma vez por dia</p> <p>6 Diversas vezes por semana</p> <p>4 Uma vez por semana</p> <p>2 Algumas semanas podem se passar. Só disponibilizamos código quando está a tarefa está pronta</p> <p>0 Geralmente tenho problemas porque muitas mudanças acontecem entre o momento em que eu faço checkout e quando tento sincronizar. Tenho que tomar cuidado na sincronização pois geralmente posso esquecer algumas coisas. O build parece quebrar frequentemente.</p>

<p><b>Desenvolvimento Dirigido por Testes</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Temos testes automatizados para cada classe de produção?</i></p> <p>10 Os testes automatizados são o design. Se apagar qualquer linha do código, algum teste ficará vermelho. O cliente roda testes de aceitação.</p> <p>8 Após pensar no design e escrever um pouco de código, nós escrevemos o teste automatizado.</p> <p>6 Nós tomamos o cuidado de escrever testes de unidade para o nosso código antes de entregá-lo para o time de teste</p> <p>4 Nós ouvimos falar de ferramentas como o JUnit, mas nunca utilizamos.</p> <p>2 Nossa fase de teste formal ao final de cada ciclo demora muito mais que o planejado porque sempre aparecem diversos bugs e correções devem ser realizadas.</p> <p>0 Nós não temos nenhum teste formal. Os clientes geralmente nos avisam se encontrarem algum problema.</p>
<p><b>Jogo do Planejamento</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Nós movemos histórias para dentro e fora do plano baseado nas necessidades atuais do cliente, mantendo os prazos fixos.</i></p> <p>10 Antes de cada iteração curta o cliente escolhe as funcionalidades mais valiosas baseado nas estimativas dos desenvolvedores. A cada manhã nós revisamos as principais histórias numa reunião em pé de 5 minutos e os desenvolvedores se voluntariam para parear e desenvolvê-las. Como sabemos que mudanças acontecem, temos vantagem competitiva porque otimizamos nosso processo para aceitar e nos adaptar a elas.</p> <p>8 Às vezes nós movemos funcionalidades para dentro e para fora do plano logo após o cliente mudar de prioridade, sempre que o time de desenvolvimento termina antes da hora ou se atrasa em alguns itens. Afinal, o produto é do cliente. Eles devem ter o que quiserem. Mudanças acontecem.</p> <p>6 Planos não devem mudar. Nós cumprimos nossas datas, mesmo quando planejadas com 1 ano de antecedência. Nós criamos diversos artefatos como, por exemplo, documentos de especificação. Nós tentamos deixá-los atualizados.</p> <p>4 Nós tomamos o cuidado de seguir o processo "cascata". Nós não começamos a escrever código antes de termos completado e revisado os documentos de design e especificação. Não começamos a testar antes do código ser entregue. Algumas vezes nós mudamos ou perdemos a data de entrega porque o cliente mudou algum requisito.</p> <p>0 Nós perdemos clientes porque dizemos que eles devem esperar pelo próximo <i>release</i>. Afinal, nós estamos ocupados tentando terminar o que pediram no ano passado.</p>
<p><b>Presença do Cliente</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Nós temos acesso ao nosso cliente e obtemos feedback frequente.</i></p> <p>10 Nós não consideramos uma história completa até que o cliente execute seu teste de aceitação.</p> <p>8 Nós freqüentemente interagimos com nossos clientes para mostrar protótipos e ver como eles querem mudá-los.</p> <p>6 Nós obtemos requisitos dos clientes.</p> <p>4 Os requisitos vem de algum lugar, mas não acho que venham dos clientes.</p> <p>2 Nós entregamos funcionalidades, mas nunca temos certeza se fizemos o que eles queriam. Eles provavelmente já devem ter mudado de idéia agora.</p> <p>0 Nós sabemos o que é certo. Eles usarão nosso sistema gostando ou não. Será bom para eles.</p>
<p><b>Refatoração</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Nós re-escrevemos ou re-projetamos código com "mau cheiro" para facilitar o desenvolvimento de novas funcionalidades.</i></p> <p>10 Nós geralmente usamos a refatoração como ferramenta para tornar nosso design mais flexível e facilitar a implementação de mudanças.</p> <p>5 Nós realizamos alguma limpeza no código de tempos em tempos.</p> <p>0 Nós temos muito código antigo que já passou por diversas mudanças ruins. Nós temos medo de alterá-lo. Nós recusamos requisitos novos pois o código funciona como especificado e não pode ser alterado.</p>
<p><b>Design Simples</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Nós mantemos o design simples agora para que possamos alterá-lo diante das mudanças.</i></p> <p>10 Nós executamos refatorações freqüentes. Nós não tentamos prever quais funcionalidades podem ser utilizadas no futuro. Generalizações precoces não são consideradas.</p> <p>8 O design é limpo. Ele faz somente o necessário de uma forma simples.</p> <p>6 Nosso design é intuitivo, com poucos pontos de melhoria.</p> <p>4 Nós projetamos uma funcionalidade inteira com orgulho. Um produto que faz tudo que as pessoas precisam.</p> <p>2 Nós temos grandes peças de código não-terminado que eventualmente são jogados fora.</p> <p>0 Eu pensei em escrever esse <i>framework</i> caso alguém precise no futuro. Nunca se sabe.</p>

<p><b>Padronização do Código</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Você segue uma padronização no código que permite alterações em qualquer parte do sistema?</i></p> <p>10 Nós temos um padrão, o seguimos e treinamos novas pessoas a seguirem. Aliás, são padrões da indústria também.</p> <p>8 Nós temos boa parte do padrão documentada, e as pessoas geralmente o seguem.</p> <p>6 Nós fazemos a mesma coisa em alguns casos, porém outros são tratados diferentemente. Nossas chaves são colocadas no mesmo lugar, mas em trechos de tratamento de erros estão diferentes.</p> <p>4 Nós temos diversos padrões e cada um segue o seu.</p> <p>2 Nós não temos padrão.</p> <p>0 Como você ousa me dizer o que fazer?</p>
<p><b>Propriedade Coletiva do Código</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Qualquer um pode alterar qualquer parte do código. Nós não precisamos esperar o especialista.</i></p> <p>10 Nós mudamos código em qualquer área regularmente. Você não consegue perceber pois nosso código parece o mesmo.</p> <p>8 Nós mudamos código em qualquer área regularmente.</p> <p>6 Nós mudamos código alheio, mas geralmente dividimos as tarefas de acordo com a especialidade de cada um.</p> <p>4 Nós podemos consertar isso se for necessário.</p> <p>2 Nós teremos que esperar ele voltar das férias.</p> <p>0 Eu bloqueio os arquivos do meu código e mantenho-os bloqueados.</p>
<p><b>Metáfora ou Sistema de Nomes</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Você tem uma convenção de nomes para seus objetos?</i></p> <p>10 Posso dizer o que o código de qualquer programador faz somente lendo os nomes, sem olhar os comentários. Geralmente pensamos nos mesmos nomes para as coisas. Quando estamos discutindo sobre o design nós usamos uma metáfora comum para o sistema.</p> <p>8 Eu gosto dos nomes no sistema. Eles fazem sentido para mim.</p> <p>6 Eu posso seguir a nomenclatura, com alguma ajuda dos comentários.</p> <p>4 Os nomes são confusos. Você realmente precisa entender o que o código faz. Não tenho certeza qual nome dar às classes.</p> <p>2 Você precisa conhecer o histórico, o nome é um fator, mas os métodos fazem coisas diferentes pois evoluíram independentemente.</p> <p>0 Por que eles usaram esse nome? Não posso entender as abreviações.</p>
<p><b>Ritmo Sustentável</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>As pessoas trabalham num ritmo que é produtivo e sustentável no longo prazo?</i></p> <p>10 Nós trabalhamos num ritmo constante e confortável. Nós corremos somente quando necessário, somente em poucas ocasiões.</p> <p>8 Algumas vezes estou muito ocupado, outras muito entediado.</p> <p>6 Sempre que temos uma data de entrega precisamos entrar num modo de "correria".</p> <p>4 Já faz alguns meses que estamos pedindo comida para a janta. Parece que sempre será assim.</p> <p>2 Já precisei cancelar aulas e férias mais de uma vez.</p> <p>0 Não tenho tempo para preencher esse formulário.</p>
<p><b>Lições Aprendidas</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Você pára periodicamente para avaliar formas de melhoria?</i></p> <p>10 As pessoas geralmente aparecem com idéias novas. Essas idéias são implementadas. Nós compartilhamos novas técnicas com pessoas de outra equipe. Fazemos retrospectivas periodicamente.</p> <p>8 Nós pensamos no que deu errado, o que deu certo e propomos mudanças para o time.</p> <p>6 Nós temos idéias mas elas parecem desaparecer e nunca são implementadas.</p> <p>4 Nós repetimos os mesmos erros.</p> <p>0 Nós nunca paramos. Tenho reclamações que parecem nunca serem levadas em conta. Obviamente ninguém se importa.</p>
<p><b>Tracking</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Existem diversos gráficos e informações espalhadas pelas paredes que nos ajudam a entender o andamento do projeto.</i></p> <p>10 Atualizamos as informações diariamente e descartamos os gráficos que não trazem informações relevantes. Eles nos ajudam a entender o andamento do projeto e identificar pontos de melhoria.</p> <p>8 Temos alguns gráficos interessantes que são atualizados semanalmente.</p> <p>6 As informações na parede são atualizadas ao final de cada <i>release</i>.</p> <p>4 Os gráficos estão desatualizados e ninguém mais se importa com o que está na parede. Precisamos terminar o trabalho para entregar o sistema no prazo.</p> <p>2 Não sei qual a utilidade dos gráficos na parede. Eles parecem não estar relacionados ao meu trabalho diário. Eu acho que se fossem retirados ninguém iria perceber.</p> <p>0 Não temos nenhum gráfico na parede. Preferimos guardar informações importantes em arquivos e documentos que ficam armazenados em nosso repositório central. Quem tiver interesse pode atualizá-los ou consultá-los.</p>

<p><b>Nota geral para XP</b></p> <p>Atual: _____</p> <p>Ideal: _____</p>	<p><i>Como você avalia sua implementação geral das práticas de XP?</i></p> <p>10 Eu sou <i>coach</i> de XP, escrevo livros e faço apresentações em conferências.</p> <p>8 Somos um bom exemplo de time de XP.</p> <p>6 Nós geralmente mudamos um pouco as práticas na nossa rotina diária. Nós lemos sobre XP, mas como vidas ou muito dinheiro dependem do nosso software e como temos um time muito grande, usamos métodos formais com muita documentação e reavaliações periódicas.</p> <p>4 Eu não sei o que é XP, mas as perguntas desse questionário parecem intrigantes. Talvez possamos tentar algumas delas em algum momento.</p> <p>2 Não tenho certeza o que é XP, mas me parece algo ruim.</p> <p>0 Não acredito que o desenvolvimento de software deva estar amarrado a um "processo", ou, nós sempre fizemos negócios da mesma forma. Não vejo necessidade de mudar agora.</p>
--	--

**Observações Finais / Sugestões:**

---



---



---



---



---



---