

Metodologia de desenvolvimento ágil Extreme Programming – XP

Em meados de 1990, Kent Beck procurou formas mais simples e eficientes de desenvolver softwares (identificando o que tornava simples e o que dificultava o desenvolvimento de software). Em março de 1996, iniciou um projeto com novos conceitos, então nasceu a metodologia XP – eXtreme Programming.

“Trata-se de uma metodologia de desenvolvimento de software ágil para equipes pequenas e médias desenvolvendo software com requisitos vagos e constante mudança.” Kent Beck.

Apesar de ser reconhecida como uma metodologia Ágil, ela é na verdade um conjunto de boas práticas que tem o propósito de auxiliar o desenvolvimento de sistemas. Tendo como finalidade principal o desenvolvimento a documentação é muito menor que em outras metodologias, a XP é recomendada para pequenas e médias equipes envolvidas em projetos com requisitos vagos e em mudanças constantes.

O XP se mantém sobre quatro valores básicos e algumas práticas, sendo que essas podem variar na quantidade conforme a bibliografia.



OS PRINCÍPIOS BÁSICOS:

Comunicação:

Um dos pontos mais importantes da metodologia, e o que garante que não existam lacunas em processos e problemas entre cliente, equipe e fornecedor.

Feedback:

É um complemento ao valor da comunicação seu objetivo é garantir que sempre haja retorno de informações entre os membros da equipe e também para o cliente, seja através de e-mails, telefones ou outros meios disponíveis.

Simplicidade:

É recomendada para todo o projeto, desde o levantamento de requisitos até implantação do sistema.

Coragem:

Que seria o estímulo necessário para informar quando algo que mudaria o rumo do projeto precisa ser feito, como por exemplo: algo solicitado pelo cliente não tem como ser feito, ou a entrega do projeto precisará ser adiada em virtude da mudança de requisitos ou mesmo que um código que funciona precisará ser refatorado por um motivo outro.

AS PRÁTICAS:

Cliente presente:

O XP sugere que o cliente esteja no dia-a-dia do projeto, acompanhando os passos dados pelos desenvolvedores, onde sua ausência poderia causar sérios riscos ao projeto. Enquanto os desenvolvedores implementam as histórias passadas pelos clientes, o mais eficaz é dialogar com o cliente enquanto as implementações são feitas.

Releases e Iterações:

Releases se trata de mostrar em curtos espaços de tempo um conjunto de funcionalidades bem definidas e que representam algum valor ao cliente. Um projeto XP pode ter um ou mais releases no seu ciclo de desenvolvimento. Uma iteração é um espaço ainda menor de tempo dentro do release. Contém um conjunto de histórias a serem implementadas, com duração de uma a três semanas, onde no final das mesmas possa ser enviado ao cliente para ele validar as implementações e dar um feedback a equipe.

Padrões de codificação:

Tem o propósito de facilitar a integração com módulos construídos separadamente e se apoia nas práticas de posse coletiva e programação em par por facilitar o trabalho em equipe.

Jornadas de trabalho fixas (40 horas semanais):

Apesar de já ser bastante comum no Brasil, foi uma pratica incluída na metodologia após ser constatado que uma mente descansada trabalha melhor. Portanto, adiar algo para não trabalhar em excesso pode fazer com que essa etapa adiada venha a ser melhor desenvolvida.

Integração continua:

Desenvolver grandes processos separadamente para então integra-los quase sempre é mais trabalhoso que desenvolver, testar e integrar pequenos processos. Com isto dito, a XP tem como pratica a integração continua, que se baseia exatamente neste conceito. Após algum código estar finalizado e testado, ele deve ser integrado ao sistema maior e testado, desta forma o código principal estará sempre em evolução e funcionando de acordo com a etapa em que o projeto se encontra.

Programação em par:

Uma das práticas mais lembradas ao mencionar a metodologia em questão, mesmo não sendo exclusiva dela. A programação em par é utilizada com o conceito de que as

vezes uma pessoa está tão focada em algo que acaba deixando passar despercebido algo que poderia ser melhorado ou que possa prejudicar o projeto como um todo.

Reuniões em pé / Reuniões Diárias:

Reuniões rápidas para não perder foco nos assuntos, apenas abordar tarefas realizadas e a realizar da equipe.

Refatoração:

É um processo que permite a melhoria continua da programação, com o mínimo de introdução de erros e mantendo a compatibilidade com o código já existente. “Refabricar” melhora a clareza (leitura) do código, divide-o em módulos mais coesos e de maior reaproveitamento, evitando a duplicação de código-fonte.

Metáfora:

Procura facilitar a comunicação com o cliente, entendendo a realidade dele. Isso seria a conversa com o cliente, pondo suas palavras de um não conhecedor de desenvolvimento de software, em palavras legíveis aos programadores que irão realizar o projeto.

BIBLIOGRAFIA:

- <http://hiperbytes.com.br/artigos/metodologia-xp-extreme-programming-boas-praticas-da-xp/>
- <http://www.devmedia.com.br/conceitos-basicos-sobre-metodologias-ageis-para-desenvolvimento-de-software-metodologias-classicas-x-extreme-programming/10596/>
- http://www.cin.ufpe.br/~synergy/epf/karua_process/guidances/supportingmaterials/boas_praticas_xp_B1D9CC77.html

Acessados em 27 de março de 2016