

## Engenharia de Software Método XP

Atividades: 01. Sintetize a metodologia XP através de um mapa mental ou um resumo de uma lauda e faça um post no instagram da disciplina.

# Método Extreme Programming

### -Origem do XP:

Em síntese é um tipo de metodologia ágil proposta por Kent Beck consequência do Manifesto Ágil feito um livro com a versão mais acessível para o público em 2004. O seu objetivo principal é mais código e menos burocracias desenvolvendo um código/sistema com alta qualidade, simplicidade e velocidade no prazo.

### -Um pouco da organização e o cliente:

O XP baseia-se num trabalho com o **cliente**. Mútuo com o usuário eles utilizam de menos documentações e usos de “user stories” contendo uma linguagem padronizada enxuta do desejo do cliente ao qual ajuda na programação e entender como vai ser o software e o que deve se melhorar através de variáveis, estimativas e organizações como **Post Its** dando prioridade às tarefas que foram propostas pelo usuário. Existe uma ordem de prioridade distribuída entre essencial (“o que depende de várias funções e “sub programas ””) e os não essenciais são trabalhados no decorrer do projeto que é pensado em cima da leitura e análise das variáveis dos “user stories”.

### -Coragem ao programar:

Após o planejamento que é de modo iterativo incremental (deposta da análise e priorização dos “user stories”)

na hora da implementação a equipe deve ter **coragem**, uma vez que se a função não for essencial ou não bater tal requisito com o objetivo principal, ou acaba mais atrapalhando do que auxiliando na implementação (possíveis ferramentas desnecessárias sem se preocupar no essencial) os programadores deverão refazer o trabalho, saber pedir ajuda falar que não sabe também faz parte do processo. Todos os programadores têm que saber um pouco do projeto e devem estar dispostos a ter reuniões e saber lidar com críticas construtivas caso não bata o nível de qualidade ou satisfaça a funcionalidade proposta.

### -Entendimento final geral do XP:

A organização ela ocorre inicialmente no projeto quebrando-o em diversas partes e estabelecendo padrões de programação, aos quais cada dupla de programador que durante tempos periódicos trocam de função (Um analisa e pesquisa referências para o código e o outro efetivamente implementa) irá desenvolver de acordo com a função proposta e fazer refatorações ou seja testes nos códigos que possibilitam o encontro de erros e assim melhorar a técnica, simplicidade e legibilidade do código.

03. Prática Técnica XP - Criar um repositório no github e descrever as práticas da metodologia XP aplicadas ao desenvolvimento do projeto do estudo de caso.

O projeto é um aplicativo de investimentos em ações, inspirado na corretora Rico pertencente ao Grupo XP, onde é possível que o usuário veja todas as ações disponíveis para compra na página principal, bem como as ações já adquiridas na sua carteira. É possível comprar uma nova ação, e comprar ou vender ações já adquiridas anteriormente. A aplicação deve conter uma página da conta digital, onde é possível fazer depósitos e saques em sua conta.

**1. Conceituação do Projeto - objetivo, participantes envolvidos, os papéis assumidos por cada um. Demonstrar como tudo começa em um projeto que utiliza XP bem como a proposta do cliente e as condições do aceite.**

**2. Planejamento Inicial: Uma vez feito um contrato com o cliente, a equipe que utiliza XP deve se reunir com o mesmo para conhecer melhor o sistema. Assim, esta documentação deve demonstrar como coletar e documentar requisitos, fazer as estimativas, dar um prazo inicial e dividir os requisitos.**

**3. Planejar a iteração: Sempre que uma iteração nova começa na XP é necessária uma reunião de planejamento para distribuir as atividades entre os desenvolvedores. Apresentar como esse processo é realizado e quais atitudes devem ser observadas.**

**4. Implementando a iteração: Agora os desenvolvedores apresentarão as práticas da XP utilizadas para desenvolver o projeto. Demonstrar o dia a dia dos desenvolvedores e práticas como reuniões em pé, TDD e Programação em Par.**

**5. Entrega da iteração: Ao final da iteração muito código já foi desenvolvido, então é hora de fazer o fechamento antes de iniciar a**

**próxima iteração. A equipe deve se reunir para fazer observações e destacar as lições aprendidas.**

**6. Entrega da release: Por fim, após várias interações, a release é concluída. Apresentar a última etapa da XP que é a entrega da release, onde cliente e desenvolvedor se reencontram para testes e para as atividades de aceitação do projeto**

**Relato do Projeto de Investimentos:** Composto por uma equipe pequena de duas pessoas aos quais apresentaram conhecimento especializado para gerir todo o projeto de maneira ao qual apresente qualidade e velocidade tem como inicialização **a primeira conversa com o cliente via presencial** e foi pedido algumas “estórias do cliente” para levarmos como parâmetro.

Após o lançamento das diversas “estórias do usuário” ouve uma **análise de requisitos** estipulados seguindo em consideração quais são as funcionalidades principais que o cliente quer (evitando código e ferramentas desnecessárias). Concomitante ao processo, passamos para a **estruturação e separação** do programa principal **em Post its** que mostram as funcionalidades que serão abordadas ao decorrer do projeto e sua devida prioridade. Após montar e organizar o projeto com as devidas permutas de funções separadas identificamos **as principais características do software**, apresentamos para o cliente para ter certeza do que usar ou não para finalmente **começar em dupla a implementar**.

-O Processo de repetição da implementação do código durante todo o projeto: **Começamos com um código simples chamado “spikes”** levando em consideração um requisito do Post it e sem finalidade direta em entrar para o código principal. Durante o processo enquanto um da equipe implementa outro vê e analisa uma forma de transformar esse código mais “elegante” após a implementação dessa funcionalidade caso seja essencial no programa principal um dos **programadores explicam para o cliente para ver se satisfaz o requisito** implementado e caso passe ai sim vai para a análise e direto para o programa principal onde está separadas por esses “sub códigos” que **compõem a Release** que nada mais é uma versão do software que está em produção e sofre alterações.

Ao decorrer da Release a equipe se compõem em reuniões para a organização e decisão final de que funcionalidade atribuir, fazer ou descartar. Quando o **software apresentar valor agregado** a missão da equipe é **realizar testes automatizados e refatorações** junto ao cliente caso dê errado a equipe refaz ou verifica se voltam alguma versão para a continuação do projeto(analisando o nível de prioridade). **Na refatoração** ela troca nome de variáveis escalona e coordena padrões de linguagem **deixando o código do projeto enxuto, simples e flexível** para novas mudanças e implementações.

-Situação do projeto:

De maneira resumida as releases feitas, analisadas e aprovadas são mandadas para o código principal e sempre **fazemos caso haja uma alteração no código principal uma integração no dia**. Isso faz com que haja num mesmo repositório (onde está guardado uma funcionalidade ou o próprio código) ter **várias versões de integração** evitando assim problemas futuros.

Enfim após todas essas versões e iterações de modo iterativo incremental onde sofre alteração e contato direto com o cliente validando ou não a funcionalidade, seja com “estórias de usuário”, “ligações” chegamos a um consenso e finalizamos a página apresentada no git.

