UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Análise e Projeto de Software

Universidade Evangélica de Goiás

Curso de Engenharia de Software

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Análise e projetos de Software: Modelo Ágil

Introdução as Metodologias Ágeis

Os métodos ágeis, desde seu surgimento, na década de 1990, têm sido apontados como uma alternativa aos modelos clássicos ou tradicionais para o desenvolvimento de software. Dessa forma, discutem-se, há muito tempo, as diferenças e as semelhanças entre essas duas abordagens, e algumas características têm sido apresentadas para definir suas aplicações aos processos de software.

As abordagens tradicionais são consideradas pelos seguidores dos métodos ágeis como soluções complexas, pesadas ou fortemente calcadas no planejamento. Com certeza, a prática mostra que elas nem sempre conseguem atender aos projetos em que há muitas mudanças ao longo do desenvolvimento e quando não existe muita clareza nos objetivos e soluções que deverão ser implementados.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Introdução as Metodologias Ágeis

Como atender ao ambiente extremamente dinâmico que as organizações estão vivendo e responder rapidamente às demandas e expectativas do mercado e dos clientes? É para responder a essas indagações que surgiram os métodos ou modelos ágeis, com suas propostas inovadoras na forma de se construir software.

Uma definição consagrada de método ágil ou desenvolvimento ágil de software é: conjunto de atividades, métodos ou processos de desenvolvimento de software. Outra definição muito comentada seria: o desenvolvimento ágil, tal como qualquer processo de software tradicional, fornece uma estrutura conceitual e um conjunto de práticas para projetos de software (COSTA et al., 2013; AMBLER, 2004)

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

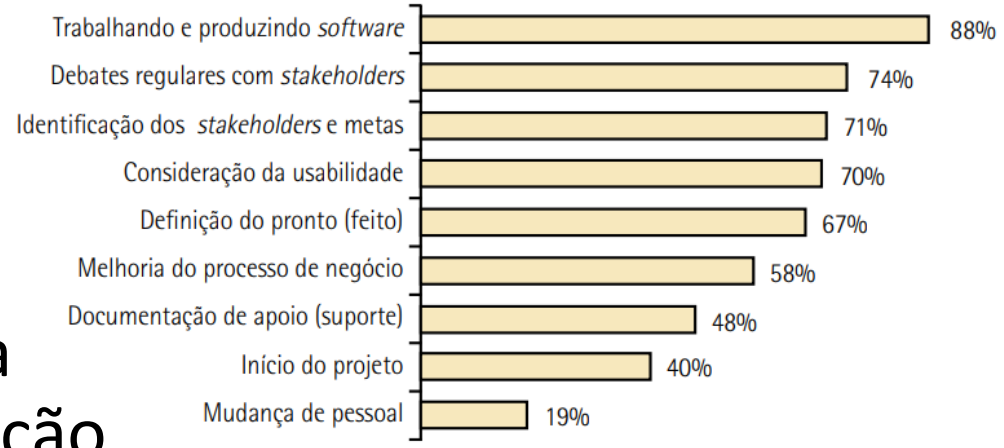
Introdução as Metodologias Ágeis

Os métodos ágeis originaram-se a partir da década, de 1990 e a principal razão apontada era o fato de que o modelo em cascata ou modelo clássico era visto como extremamente burocrático, lento e contraditório em relação à forma usual de os engenheiros de software realizarem seus trabalhos com eficiência. Têm muito em comum com as técnicas de desenvolvimento rápido de aplicação, RAD, da década de 1980, sugeridas por James Martin, Steve McConnell, entre outros autores. Mas o que propunha o desenvolvimento RAD?

- o RAD usa o mínimo de planejamento em favor de uma rápida prototipagem;
- o planejamento é intercalado com a escrita do software;
- geralmente, a falta de um pré-planejamento extensivo permite que o software seja escrito muito mais rapidamente;
- torna mais fácil aceitar mudanças de requisitos ao longo do processo;
- é uma metodologia que inclui o desenvolvimento iterativo e a prototipagem de software;
- é uma combinação de várias técnicas estruturadas, especialmente, da engenharia de informação, que é dirigida por dados, com técnicas de prototipagem para acelerar o desenvolvimento de sistemas;
- as técnicas estruturadas e a prototipagem são usadas para a definição dos requisitos dos usuários e para o design do sistema final.

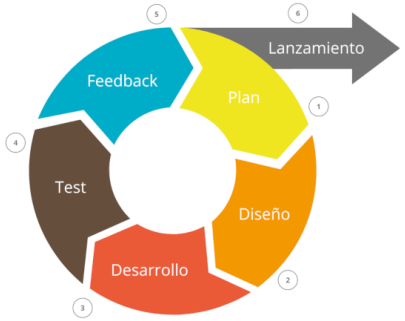
UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Introdução as Metodologias Ágeis



Recentemente, a empresa americana Ambysoft fez uma pesquisa com a intenção de verificar como o mercado vê de que forma os times ágeis agregam valor ao cliente, e os resultados são apresentados na figura a seguir.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO



Introdução as Metodologias Ágeis

O objetivo da pesquisa era explorar como as **equipes ágeis** adotaram os critérios ágeis, e as respostas da figura anterior, especificamente, questionavam como os times estão agregando valor ao cliente ou interessado.

Os resultados mostram, pelos percentuais, que os times consideram que o fator mais importante na entrega de valor ao cliente é produzindo software (88%). Todavia, outros fatores também foram bem-avaliados, como o debate regular com o cliente, a identificação dos objetivos dos clientes etc.

É interessante notar que poucos consideram que o início do projeto e mudanças de pessoal são fatores relevantes na entrega de valor ao cliente.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Introdução as Metodologias Ágeis

Outras quatro questões foram colocadas na pesquisa:

1) A equipe está validando o seu próprio trabalho?

2) A equipe trabalha em estreita colaboração com os seus *stakeholders*?

3) A equipe se organiza?

4) O time melhorou seu processo?

Essas questões foram embasadas nos princípios dos métodos ágeis.

Introdução as Metodologias Ágeis

- A história dos métodos ágeis se inicia formalmente em fevereiro de 2001, quando membros proeminentes da comunidade de software se reuniram em Snowbird (The Lodgeat Snowbirdski Resort, em Utah) e adotaram o nome **métodos ágeis**.
- De acordo com Pressman (2006, p. 52), “nasceu o Manifesto Ágil, documento que reúne os princípios e as práticas desse paradigma de desenvolvimento”.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Introdução as Metodologias Ágeis

- Uma das observações mais importantes, ainda, na atualidade, é a que diz que os métodos ágeis são caracterizados como o oposto de metodologias guiadas pelo planejamento, ou denominadas disciplinadas ou preditivas.
- O manifesto contém quatro valores fundamentais:
- os indivíduos e suas interações, mais que procedimentos e ferramentas;
- o funcionamento do software, mais que documentação abrangente;
- a colaboração com o cliente, mais que negociação de contratos;
- a capacidade de resposta a mudanças, mais que seguir um plano preestabelecido.

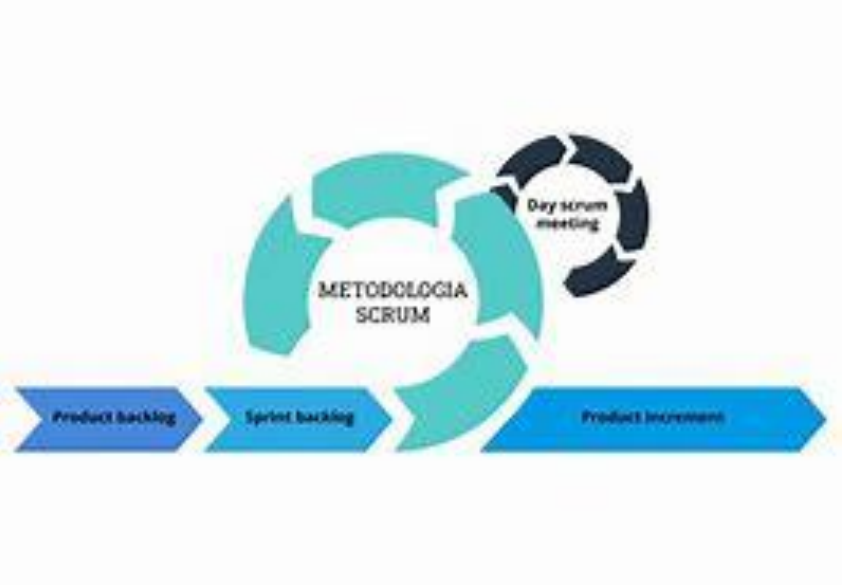
UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Introdução as Metodologias Ágeis

- Esses quatro valores, até os dias de hoje, provocam muitas discussões calorosas, e muitos especialistas consideram praticamente impossível a aplicação prática de todos esses valores na sua essência. Entretanto, a maioria dos métodos ágeis tenta minimizar o risco do desenvolvimento de software ou batalhar para eliminar a “crise do software” da seguinte maneira:

- usando ciclos curtos de tempo, chamados de iteração, sprints etc. que vão de poucos até, no máximo, 30 dias;
- cada iteração ou sprint, ou ciclo, é como um projeto de software em miniatura e inclui todas as tarefas necessárias para implantar o mini incremento da nova funcionalidade:
  - planejamento, análise de requisitos, design, codificação, teste e documentação;
  - não necessariamente documentados totalmente.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO



Introdução as Metodologias Ágeis

métodos ágeis enfatizam a comunicação em tempo real, preferencialmente, face a face, no lugar de documentos escritos;

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Introdução as Metodologias Ágeis

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

A partir dos valores do Manifesto Ágil, também foram definidos os chamados Princípios dos Métodos Ágeis, que são:

- garantir a satisfação do cliente/usuário, entregando rapidamente, continuamente e adiantadamente softwares com valor agregado e funcionando;
- softwares funcionando são entregues frequentemente (em dias, semanas, em vez de meses); — softwares funcionais são a principal medida de progresso do projeto;
- até mesmo mudanças tardias de escopo no projeto são bem-vindas;
- cooperação constante (diariamente) entre pessoas que entendem do negócio e desenvolvedores; — bons projetos surgem por meio de indivíduos motivados, em uma relação de confiança;
- simplicidade de projetos;
- rápida adaptação às mudanças;
- transmissão de informações por meio de conversa face a face;
- em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e, então, refina e ajusta seu comportamento

Introdução as Metodologias Ágeis

Os métodos ágeis são, muitas vezes, confundidos com o modelo codifica-remenda no desenvolvimento de software, principalmente, em razão de:

- o método codifica-remenda ser a ausência de metodologias de desenvolvimento de software;
- essa forma de trabalho parecer semelhante à maneira pela qual as pessoas (times) utilizam os métodos ágeis

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Introdução as Metodologias Ágeis

Outro fator importante a se analisar é a aplicabilidade dos métodos ágeis que, em geral, pode ser examinada de múltiplas perspectivas:

- da perspectiva do produto: os métodos ágeis são mais adequados quando os requisitos são instáveis (mudam rapidamente);
- da perspectiva organizacional: a aplicabilidade do método ágil pode ser expressa examinando-se três dimensões-chave da organização: cultural, pessoal e de comunicação.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

O quadro a seguir mostra uma comparação do ambiente ideal para os processos de software, tanto os ágeis quanto os clássicos ou tradicionais.

Desenvolvimento ágil	Desenvolvimento orientado ao planejamento (métodos clássicos)
Baixa criticidade do *software*	Alta criticidade do *software*
Desenvolvedores seniores	Desenvolvedores juniores (equipes mistas)
Mudanças frequentes de requisitos	Baixa mudança nos requisitos
Pequeno número de desenvolvedores	Grande número de desenvolvedores
Cultura que tem sucesso no caos	Cultura que procura a ordem por meio do planejamento

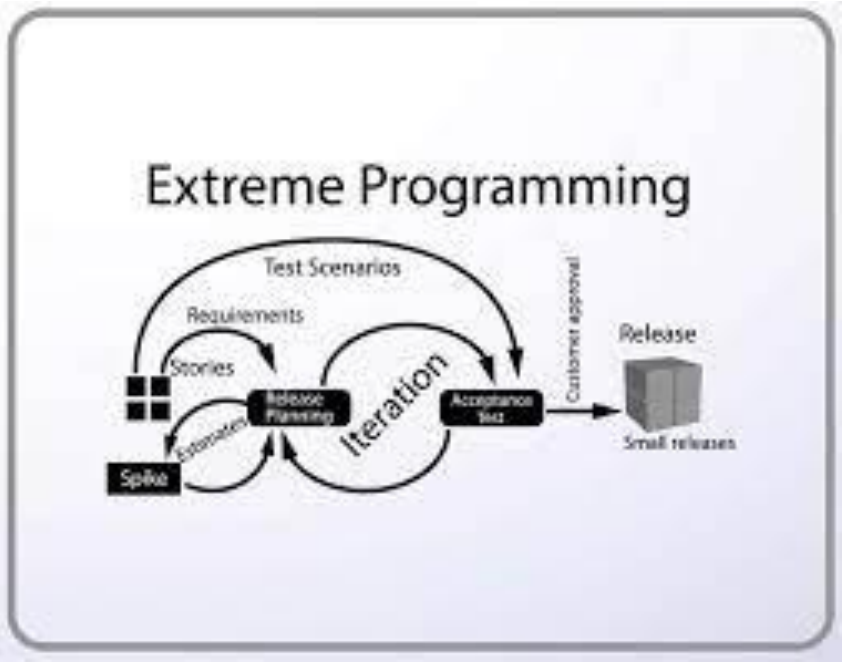
Com relação ao gerenciamento de projetos, os métodos ágeis diferem largamente dos métodos tradicionais no que diz respeito à forma pela qual são gerenciados, mas alguns métodos ágeis são complementados com guias para direcionar o gerenciamento do projeto, embora nem todos sejam aplicáveis.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Contudo, diversos autores consagrados da engenharia de software fazem críticas ao método de desenvolvimento ágil:

- esse método é, algumas vezes, criticado como codificação cowboy (termo que indica uma forma não organizada de trabalho);
- o início da programação extrema (XP) soava como controverso e dogmático, como a programação por pares e o projeto contínuo; alguns princípios dessa programação não foram fáceis de entender e aplicar à cultura vigente;

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO



O método ágil eXtremme Programming (XP)

O método ágil XP é uma disciplina leve do desenvolvimento de software, que é fortemente baseada nos seguintes princípios:

- simplicidade;
- comunicação;
- feedback;
- coragem

A disciplina ou método ágil XP é desenhada para uso em times pequenos e para quem necessita desenvolver softwares de forma rápida, num ambiente de mudanças constantes de requisitos.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

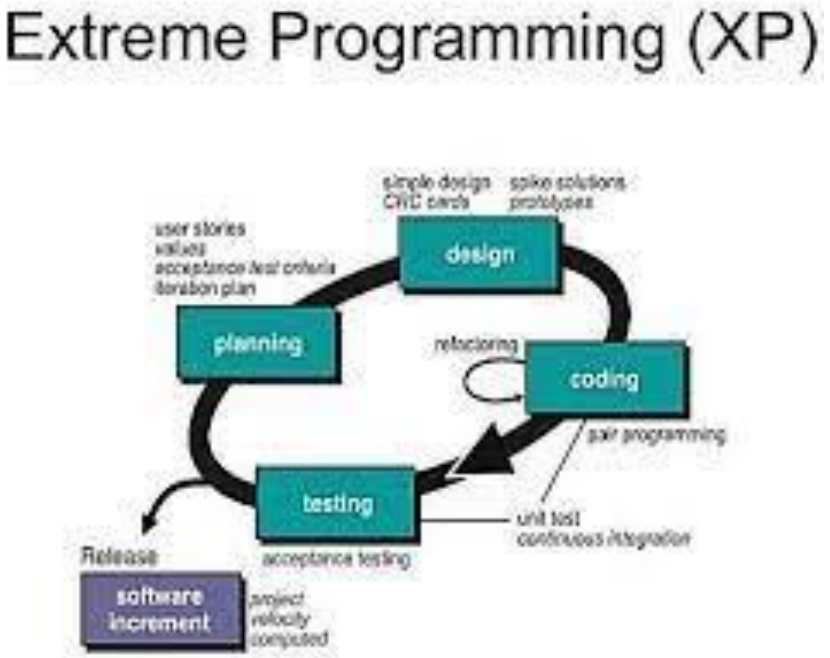
O método ágil eXtremme Programming (XP)

De acordo com Beck (2001), (considerado o “pai” ou criador do método), o sucesso metodológico da XP vem do esforço pela satisfação do cliente. O método ágil XP, quando desenvolvido por Beck, tinha como objetivos:

- a satisfação do cliente;
- o atendimento aos requisitos do cliente;
- ser fortemente focado em trabalho em times;
- manter todos voltados para criar software com qualidade

A disciplina XP enfatiza o trabalho em grupo, incluindo gerentes, clientes e desenvolvedores, todos fazendo parte do time dedicado a entregar um software de qualidade.

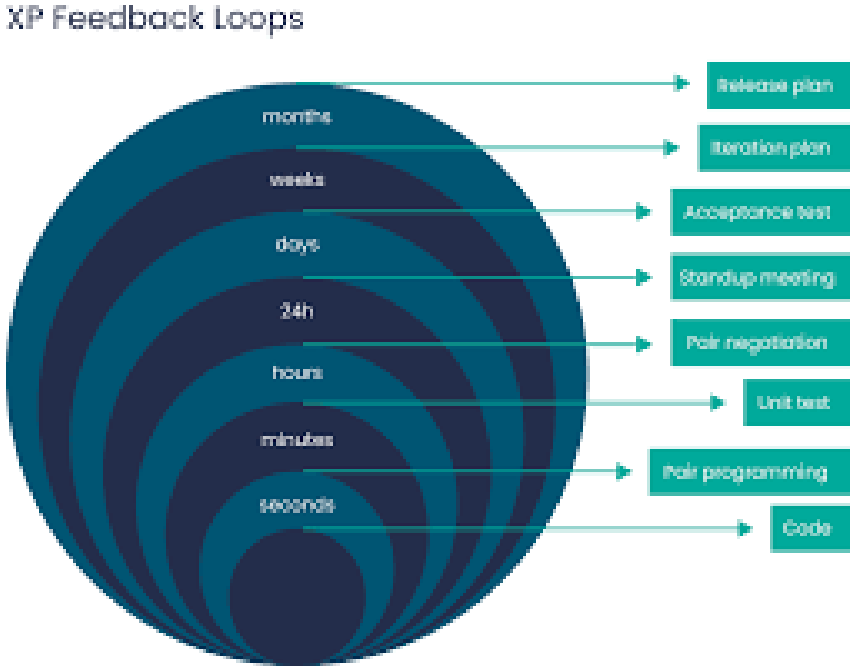
O método ágil eXtremme Programming (XP)



- Segundo o método ágil XP, existe uma limitação na definição do grupo, sendo possível um grupo de no máximo dez pessoas, incluindo gerentes e clientes.

- O método XP foi desenhado para funcionar com times pequenos, para o desenvolvimento rápido de software e para mudança constante de requisitos.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

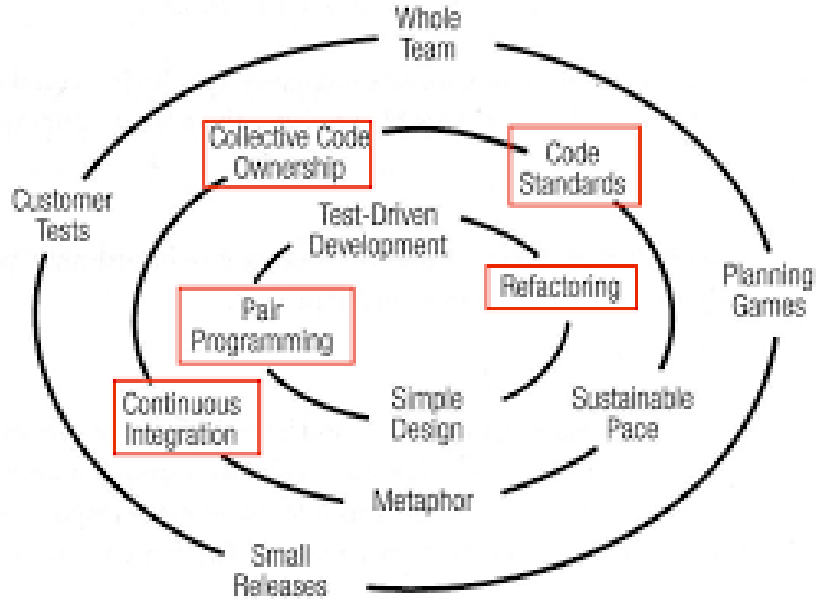


O método ágil eXtremme Programming (XP)

O método ágil XP é baseado em 12 práticas, listadas a seguir.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

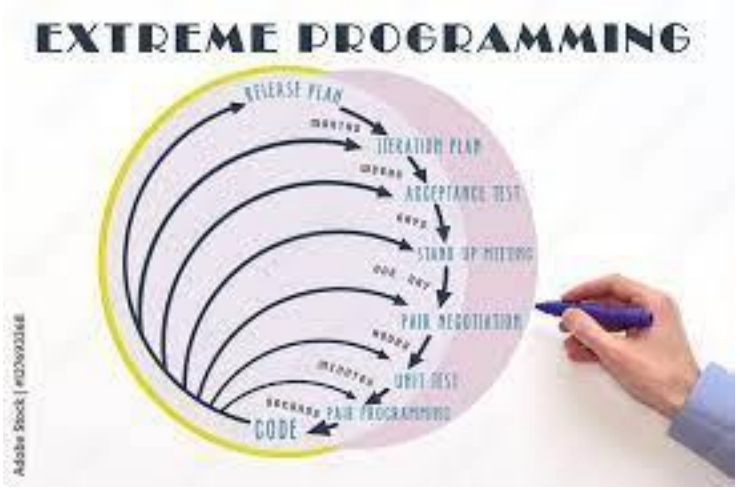
O método ágil eXtremme Programming (XP)



- P 1 (Prática 1): o processo de planejamento. O planejamento do projeto no método XP se dá por meio da técnica de reunião denominada “jogo do planejamento”.
- P 2 (Prática 2): o projeto ocorre sempre em pequenas versões. Estas são produzidas em ciclos curtos de desenvolvimento. De preferência, cada pedaço de software resultante deve ser executável e colocado em produção junto ao cliente.
- P 3 (Prática 3): metáfora. Todos utilizam nomes comuns e uma linguagem comum nos seus códigos. A padronização na escrita dos códigos é fundamental para o sucesso do método.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

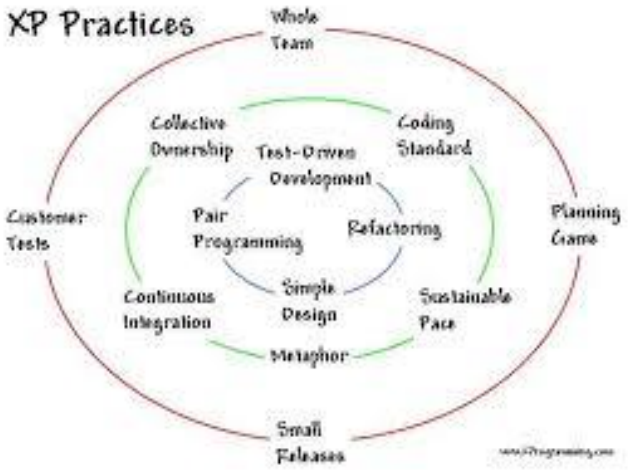
O método ágil eXtremme Programming (XP)



- P 4 (Prática 4): design simples. Não existe mais o “construir para o futuro”. O software resolve o problema de agora, sem perda de tempo estudando as futuras manutenções ou as novas funcionalidades. É uma proposta que quebra o paradigma tradicional, em que uma arquitetura deve ser trabalhada antes de se iniciar a codificação propriamente dita.
- P 5 (Prática 5): testes. A validação do software ocorre durante todo o tempo do desenvolvimento, e não no final da construção. Os códigos vão sendo construídos e testados de forma conjunta.
- P 6 (Prática 6): desenvolvimento em espiral. O método XP usa o desenvolvimento iterativo. O ciclo curto se repete até que todo o software esteja pronto.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

O método ágil eXtremme Programming (XP)



- P 7 (Prática 7): programação em pares. O código é desenvolvido por dois programadores trabalhando juntos. Essa prática faz que o código não tenha um único “dono” e resolve o problema da ausência de um dos desenvolvedores.
- P 8 (Prática 8): propriedade coletiva. Todo o código pertence a todos da equipe. Não existe “dono” do código, como nos métodos tradicionais.
- P 9 (Prática 9): integração contínua. A equipe integra o software muitas vezes ao dia, na busca da solução dos problemas de integração, tão comuns nos métodos tradicionais de desenvolvimento.

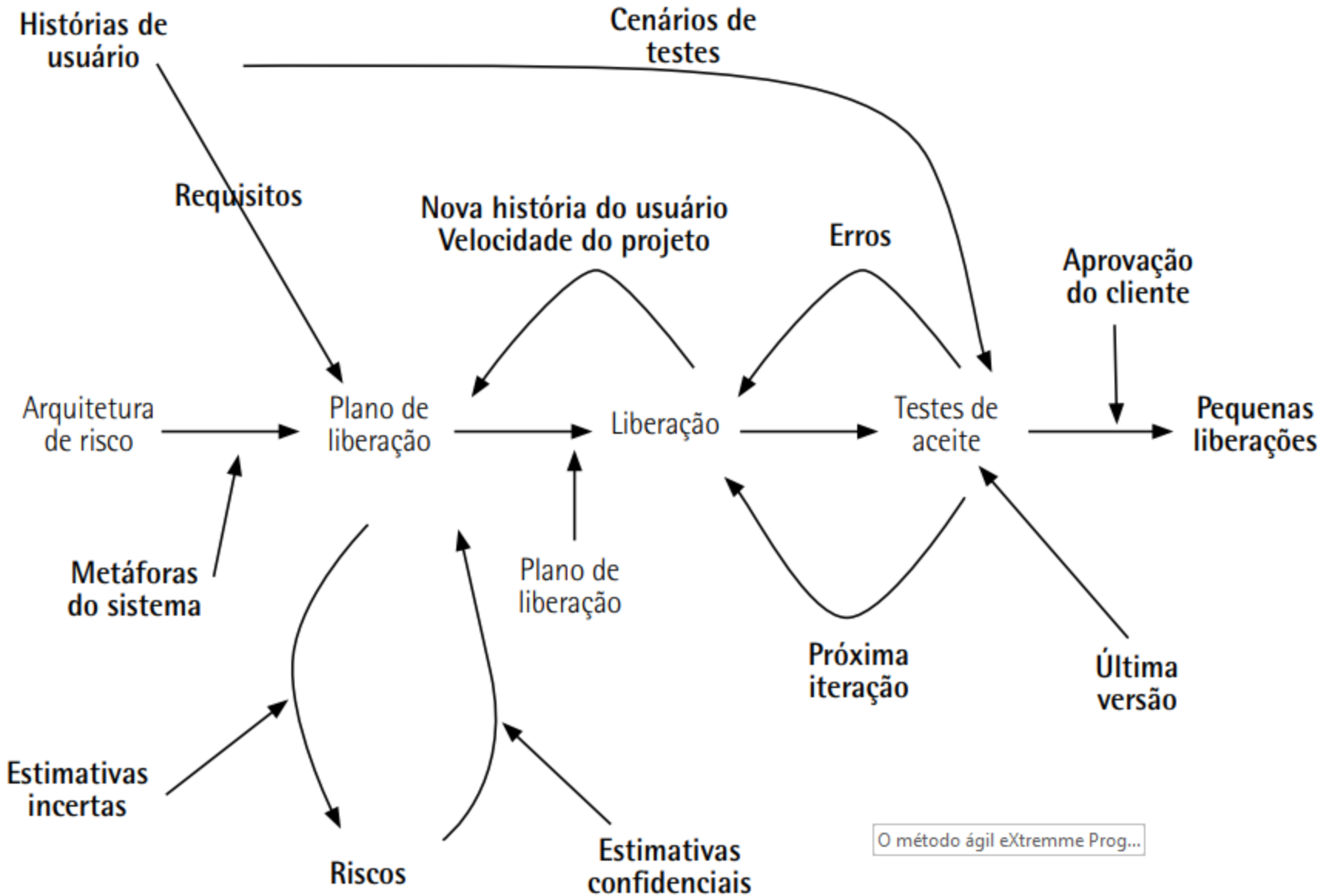
UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

O método ágil eXtremme Programming (XP)

- P 10 (Prática 10): quarenta horas de trabalho semanais. De acordo com o XP, programadores cansados cometem mais erros e prejudicam o andamento do projeto.
- P 11 (Prática 11): cliente fazendo parte da equipe. Um projeto XP é conduzido por um cliente, e decisões são definidas por ele o tempo todo.
- P 12 (Prática 12): codificação-padrão. Os códigos são escritos da mesma forma por todos os envolvidos no projeto.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

A seguir, a figura apresenta uma visão do método ágil sob o ponto de vista do ciclo de desenvolvimento.



O método ágil eXtremme Prog...

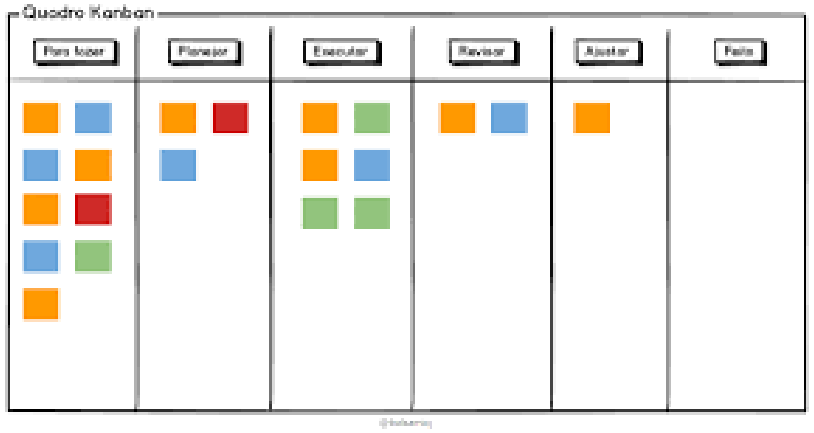
- Apesar do sucesso do método XP, principalmente, nos meios acadêmicos e nas pequenas fábricas de software, existem críticas ao método

- O autor Matt Stephens, em seu livro Extreme Programming Refactored (STEPHENS; ROSENBERG, 2003), faz uma revisão no método e apresenta as seguintes críticas:

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

- falta de estrutura e documentação necessárias: o método abriu mão, completamente, de documentos que registrassem qualquer fato a respeito do software construído; a única documentação é o próprio código-fonte;
- somente trabalhar com desenvolvedores de nível sênior: no mercado, fica muito difícil e caro montar equipes de desenvolvedores somente com alto nível de conhecimento;
- incorporar de forma insuficiente o projeto de software: como o software é construído para “agora”, quando se precisa fazer alguma manutenção, a arquitetura original nem sempre tem a estrutura necessária para as novas implementações;
- requer a adoção de mudança cultural: este é um fator importante a ser considerado quando se pretende adotar o método XP;
- Poder levar a maiores dificuldades nas negociações contratuais: como se sabe, os clientes se protegem por meio de contratos formais que não são aceitos no método XP nem fazem parte da sua estrutura, o que acaba contrastando com a realidade vivida pelas organizações.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

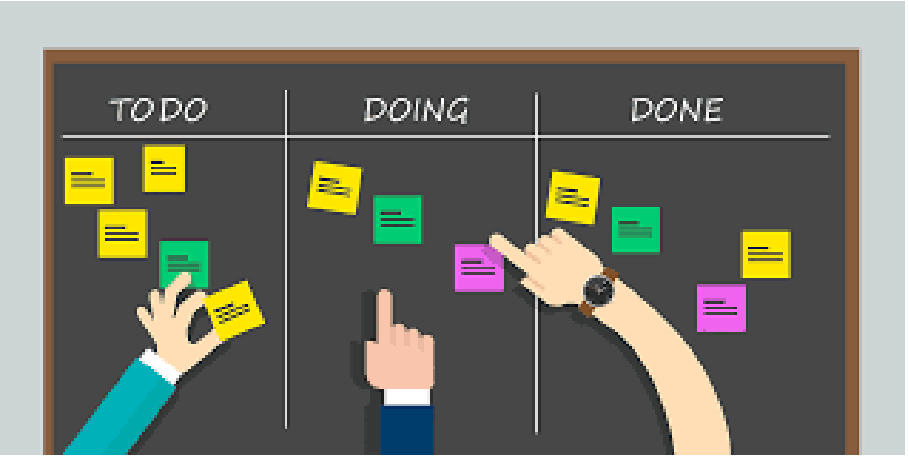


O que é Kanban

- O método Kanban é uma estrutura popular usada para implementar o desenvolvimento de software ágil e de DevOps. Ele precisa de comunicação de capacidade em tempo real e transparência total de trabalho. Os itens de trabalho ganham representação visual em um quadro Kanban, permitindo que os membros da equipe vejam o estado de cada parte do trabalho a qualquer momento.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

O que é um painel kanban?



- Um quadro Kanban é uma ferramenta de gerenciamento de projeto ágil que auxilia na visualização de trabalho, limitação de trabalho em andamento e maximização de eficiência (ou fluxo). Ele pode ajudar equipes ágeis e de DebOps a estabelecer ordem no trabalho diário. Os quadros Kanban usam cartões, colunas e melhorias contínuas para ajudar as equipes de tecnologia e serviço a se comprometerem com a quantidade certa de trabalho e terminarem tudo!

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

O que são limites de WIP?

- Em desenvolvimento ágil, os limites de trabalho em andamento (WIP) definem a quantidade máxima de trabalho que pode existir em cada status de um fluxo de trabalho. Limitar a quantidade de trabalho em andamento facilita a identificação da ineficiência no fluxo de trabalho de uma equipe. Obstáculos no pipeline de entrega de uma equipe são claramente visíveis antes de uma situação se tornar extrema.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Porque os limites de WIP são importantes?

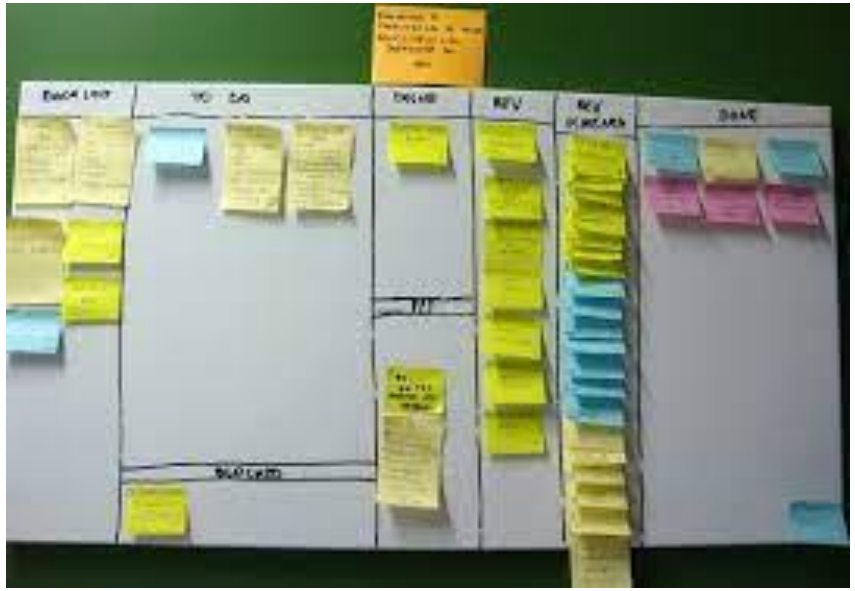
- Os limites de WIP melhoram a produtividade e reduzem a quantidade de trabalho "quase feito" forçando a equipe a se concentrar em um conjunto menor de tarefas. Em um nível fundamental, os limites de WIP incentivam uma cultura de "concluído".

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Porque os limites de WIP são importantes?

- E, o mais importante, os limites de WIP tornam os bloqueadores e os obstáculos visíveis. As equipes podem analisar itens de bloqueio para que eles sejam entendidos, implementados e resolvidos quando há um claro indicador de qual trabalho existente está causando um gargalo.

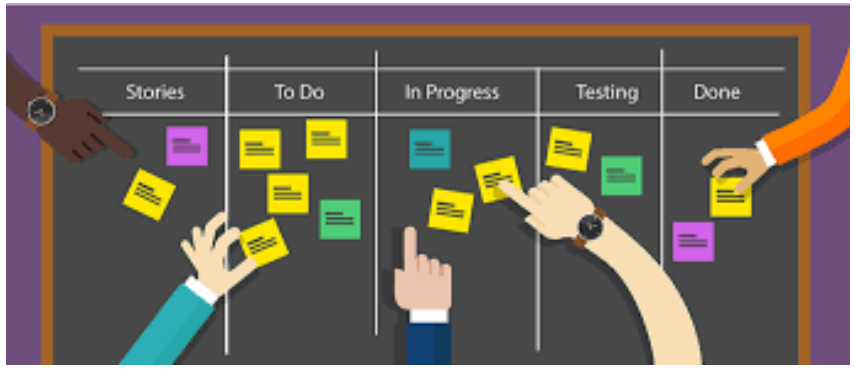
UniEVANGÉLICA
CENTRO UNIVERSITÁRIO



Porque os limites de WIP são importantes?

- Uma vez que os bloqueios são removidos, o trabalho da equipe começa a fluir de novo. Esses benefícios garantem que incrementos de valor sejam entregues aos clientes mais rápido, tornando os limites de WIP uma ferramenta importante no desenvolvimento ágil.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

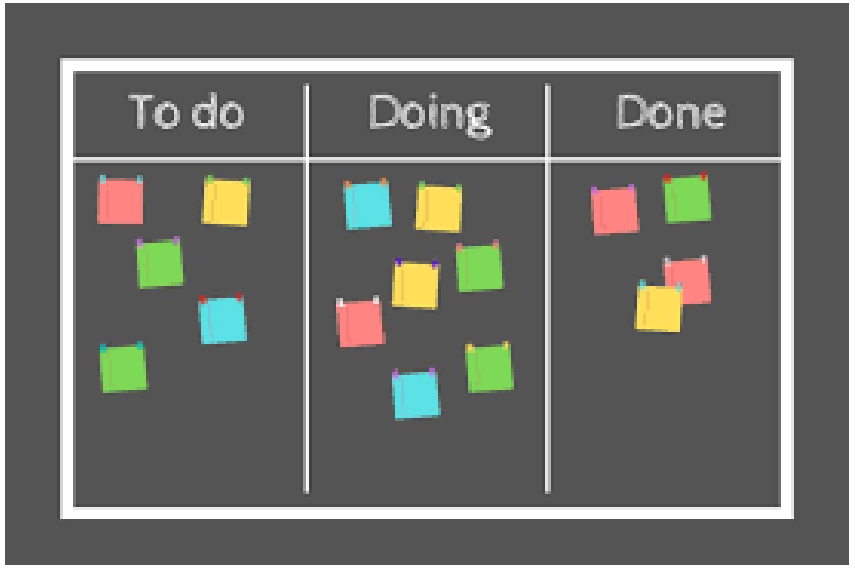


Porque os limites de WIP são importantes?

- Durante o desenvolvimento, é comum pensar: "Vou fazer uma pausa em relação a esse item enquanto começo a trabalhar em outro". Ter dois itens em aberto exige mudar de contexto entre duas coisas diferentes ou transferir trabalho entre colegas de equipe. Ir de um item para o outro não é fácil. Leva tempo e diminui o foco.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Porque os limites de WIP são importantes?



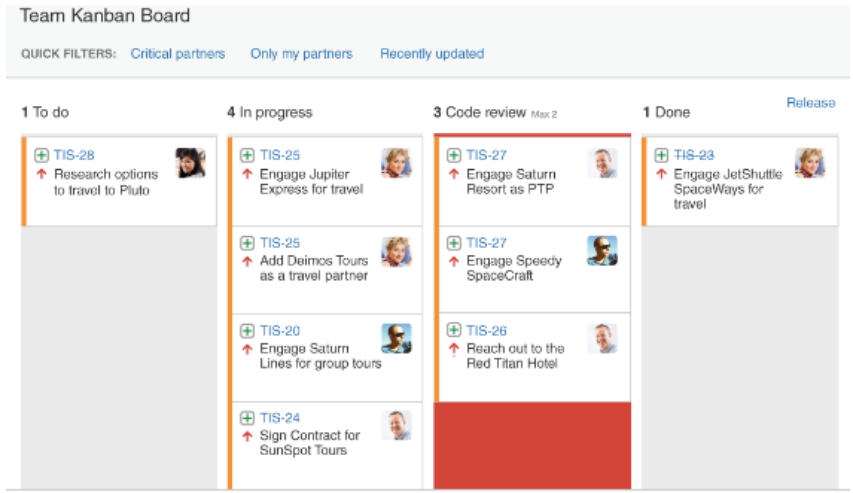
- Finalmente, os limites de WIP mostram as áreas com ociosidade crônica ou sobrecarga. Eles ajudam a equipe a ver as ineficiências no processo todo em vez de apenas em uma área específica de trabalho.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Uso dos limites de WIP em equipes ágeis

- Ao lançar um novo fluxo de trabalho, tome uma decisão em equipe para determinar os limites de WIP para cada status. A gente recomenda definir os limites de WIP depois de monitorar o número médio de itens de trabalho em cada status para alguns sprints.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO



Uso dos limites de WIP em equipes ágeis

É possível ver um quadro ágil de exemplo com os limites de WIP usados por uma equipe de desenvolvimento de software comum.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Uso dos limites de WIP em equipes ágeis

- Um limite de WIP foi definido na revisão de código. Como a coluna está excedendo seu limite, o histórico ficou vermelho.
- Como não resta nada a fazer quando um item é concluído, não há necessidade de um limite de WIP. No quadro acima, "Pendente" significa que a história foi examinada por completo pelo proprietário do produto e pela equipe. A equipe de desenvolvimento passa o trabalho de "Pendente" para "Em andamento" à medida que começam a trabalhar nos itens.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Pendente

- Como melhor prática, é importante manter trabalho suficiente no status "Pendente" para que cada membro da equipe de desenvolvimento permaneça trabalhando. Ao manter apenas histórias suficientes no estado "Pendente", o proprietário do produto não fica muito à frente quando se trata de expor os requisitos, e o programa se torna mais responsivo a mudanças.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

Em Andamento

- O status "em andamento" lista o trabalho que está sob desenvolvimento ativo.
- A meta dos limites de WIP nesse caso é garantir que todos tenham trabalho para fazer, mas ninguém esteja fazendo múltiplas tarefas. No quadro cima, o limite para os itens "em andamento" é quatro e há, no momento, três itens nesse estado.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

- Isso mostra à equipe que eles conseguem assumir mais um trabalho. Como melhor prática, algumas equipes definem o limite de WIP máximo abaixo do número de membros da equipe.
- A ideia é preparar o espaço para as boas práticas ágeis.
- Se um desenvolvedor finalizar um item, mas a equipe já estiver no limite de WIP, eles saberão que é a hora de resolver algumas revisões de código ou chamar outro desenvolvedor para ajudar na programação.

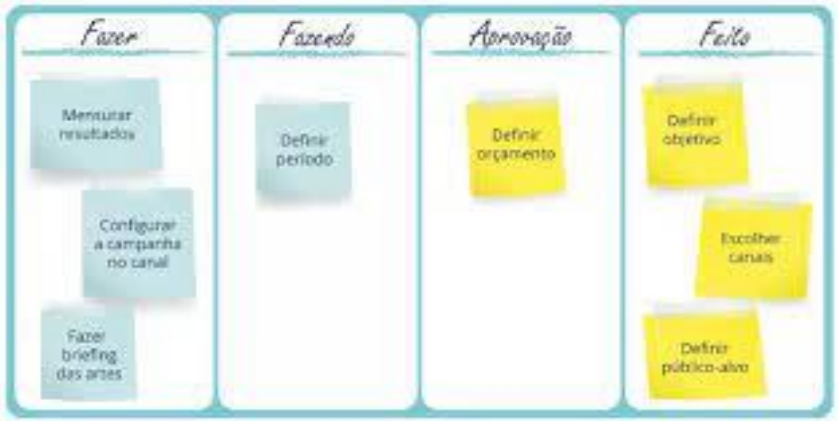
Revisão de Codigo

- O status "revisão de código" indica as histórias que foram totalmente escritas, mas precisam de revisão antes de serem mescladas na base de código. As revisões oportunas de código são uma melhor prática que estabelece qualidade, levam as inovações mais rapidamente ao mercado, tornam as mesclagens mais rápidas ao reduzir ramificações abertas e proporcionam conhecimento para toda a equipe de engenharia.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

- É necessário atuar em itens nesse estado urgentemente pelos seguintes motivos:
- O código não se torna inutilizável à medida que os membros da equipe verificam um novo código
- O desenvolvedor não perde o contexto ganho ao escrever o código original
- A função pode ser mesclada na ramificação principal para liberação
- Os limites de WIP garantem que o código não revisado não se acumule.

Quatro metas paras as equipes ágeis que usam limites de WIP



- Como em qualquer outra atividade, os limites de WIP podem parecer estranhos no começo.
- O objetivo aqui é otimizar a equipe a médio prazo, e a estranheza a curto prazo é realmente algo bom. Faz com que a equipe perceba alguns pontos problemáticos no processo.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

META 1

- Manter a consistência no dimensionamento das tarefas individuais. Ao dividir requisitos e histórias de usuário, é importante manter as tarefas individuais com não mais do que 16 horas de trabalho. Assim você aumenta a capacidade da equipe de estimar com confiança e ajuda a evitar gargalos.
- Nada atrasa uma equipe e atrapalha os limites de WIP como um grande item de trabalho obstruindo o pipeline.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

META 2

- Mapear os limites de WIP de acordo com as habilidades da equipe. O exemplo acima assume que os membros da equipe têm conjuntos de habilidades similares. Se sua equipe tem especialistas , o trabalho em limites em andamento pode ser diferente.
- Crie um status específico para o trabalho do especialista. Se houver obstáculos nesse status, use a oportunidade para mostrar aos outros membros da equipe como adicionar capacidade extra ao conjuntos de habilidades do especialista e aumentar o fluxo em toda a equipe.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

META 3

- Reduzir a ociosidade. Quando alguém da equipe tiver tempo livre, encoraje-o a ajudar nas atividades da equipe. Essa pessoa contribuirá para a produtividade geral da equipe e aprenderá algo.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO

META 4

- Proteger uma cultura de engenharia sustentável. Os limites de trabalho em andamento não foram feitos para apressar o trabalho dos desenvolvedores para evitar sobrecarga em um status específico. Eles devem dar suporte a práticas sólidas de engenharia ágil que protejam a qualidade do produto e a integridade da base de código.

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO