**Sistema de Informação**

**Engenharia de Software**

**Profº Ricardo Queiroz**

**Grupo: 6**

**Componentes:**

**Herald Freire**

**Kauano Miranda**

**Pedro Victor**

1ª SPRINT

**Estória 1. | Grupo-2 | Bruna Kelly**

**O que é processo?**

É um dialogo no qual o conhecimento, que deve se transformar em software, é reunido e incorporado ao software. O processo fornece interação entre usuário e projetistas, entre usuários e ferramentas em desenvolvimento e entre projetistas e ferramentas em desenvolvimento.

**O que são atividades de arcabouços?**

São atividades do arcabouço de processo aplicáveis a todos os projetos de software, independente de seu tamanho e complexidade.

**Quais atividades de arcabouço são aplicáveis a todos os processos de software?**

Comunicação, planejamento, modelagem, construção e implantação.

**Estória 2. | Grupo-2 | Bruna Kelly**

**Como é elaborado um projeto de software?**

**Quais sinais são comuns em projetos com falhas no futuro?**

**Estória 3. | Grupo-2 | Bruna Kelly**

**Como lidar com a imprevisibilidade em um processo?**

Consistir na adaptação de processo, permitindo alterar o projeto e as condições técnicas rapidamente. Em suma, o processo ágil deve ser adaptável.

**Defina os princípios da agilidade.**

1 – A maior prioridade é satisfazer o cliente por meio de entrega adiantada e contínua de software valioso.

2 – Acolha bem os pedidos de alterações, mesmo atrasados no desenvolvimento. Os processos ágeis se aproveitam das mudanças como uma vantagem competitiva na relação com o cliente.

3 – Entregue software em funcionamento frequentemente, de algumas semanas para alguns meses, dando preferência a intervalos mais curtos.

4 – O pessoal comercial e os desenvolvedores devem trabalhar em conjunto diariamente ao longo de todo o projeto.

5 – Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e apoio necessários e confie neles para ter o trabalho feito.

6 – O método mais eficiente e efetivo de transmitir informações para e dentro de uma equipe de desenvolvimento é uma conversa aberta, de forma presencial.

7 – Software em funcionamento é a principal medida de progresso.

8 – Os processos ágeis promovem desenvolvimento sustentável. Os proponentes, desenvolvedores e usuários devem estar capacitados para manter um ritmo constante indefinitivamente.

9 – Atenção contínua para com a excelência técnica e para com bons projetos aumente a agilidade.

10 – Simplicidade – a arte de maximizar o volume de trabalho não efetuado – é essencial.

11 – As melhores arquiteturas, requisitos e projeto emergem de equipes que se auto organizam.

12 – A intervalos regulares, a equipe se avalia para ver como tornar-se mais eficiente, então sintoniza e ajusta seu comportamento de acordo.

**Estória 4. | Grupo-2 | Bruna Kelly**

**Qual a relação entre o desenvolvimento ágil e os processos tradicionais?**

Otimização de projetos. Em métodos tradicionais de gestão, entende-se que o produto só faz sentido quando é entregue em sua totalidade, ou seja, apenas com 100% do projeto cumprido é que o cliente perceberá algum valor.

Por outro lado, métodos ágeis podem ser usados em projetos que permitem que um conjunto mínimo de funcionalidades já servirá para solucionar parte da necessidade do cliente e, ao ser entregue em parte, já representa uma diferença valorosa para ele.

Outro ponto importante entre os dois métodos é que, no ágil, quando combinado que o projeto irá entregar as funcionalidades mínimas, o cliente nem sempre tem noção do custo total do produto. Já nos métodos tradicionais, o valor é fechado junto com o escopo, o que sugere que não estão previstas alterações significativas nesse quesito durante o andamento das ações.

**Estória 5. | Grupo-2 | Bruna Kelly**

**Descreva os fatores humanos no desenvolvimento de software ágil.**

**Competência:** a equipe deve ser competente, isto é, devem ter as habilidades específicas de software necessárias e conhecer as tecnologias envolvidas em um determinado projeto ou iniciativa. Além disso as pessoas devem possuir conhecimento generalizado dos processos utilizados. Não se pode dizer que uma equipe não consegue trabalhar de forma ágil se ela não conhece os principais conceitos deste processo. Em muitas empresas, parte da equipe possui todas as habilidades técnicas necessárias, mas não conhece o processo. Isso pode ser endereçado com um simples workshop liderado por alguém que já conhece o processo.

**Colaboração:** a boa e velha habilidade de trabalhar em equipe é também fundamental. As pessoas devem colaborar entre si e com todos os envolvidos, pelo bem do projeto. Isso exige acima de tudo humildade. Até o mais sênior dos desenvolvedores tem muito o que aprender com outros colegas.

**Foco:** todas as pessoas da equipe devem estar focados em um único objetivo comum: entregar ao cliente um incremento de software funcionando, no prazo combinado. A equipe deve focar também em adaptações contínuas, sempre melhorando o processo conforme as necessidades. Lembre-se que a própria equipe deve parar de tempos em tempos (exemplo, a cada 15 dias) para refletir sobre o que está bom e o que pode ser melhorado no processo de trabalho.

**Tomada de decisão:** a equipe de desenvolvimento deve ter liberdade pra controlar seu próprio destino. Deve ter autonomia em assuntos técnicos e de projeto. É a equipe quem deve definir a melhor maneira de controlar versões de código, fazer builds, deploys, executar testes, documentar requisitos, etc. A empresa pode (e deve) sugerir boas práticas, mas no fim é a equipe (auto organizável) que irá adotar os métodos ou processos que ache melhor. As pessoas envolvidas no desenvolvimento devem saber lidar com situações conflitantes, ambiguidade e mudanças frequentes, pois são cenários que acontecem cada vez mais no dia-dia das empresas. É necessário que a equipe registre as principais lições aprendidas, o que irá facilitar no processo de melhoria contínua.

**Confiança e respeito:** a equipe deve ser consistente e precisa demonstrar a confiança e o respeito necessário para tornar a equipe forte. Lembre-se que o principal objetivo é tornar o time forte o bastante para que o todo seja maior que a soma das partes.

**Auto organização:** é a própria equipe que deve se organizar para realizar o trabalho. É preciso procurar a todo instante o que mais pode ser melhorado no processo para que ele se adeque ainda mais ao ambiente. A auto organização tem benefícios técnicos, mas é muito importante para melhorar a colaboração. A equipe seleciona quanto trabalho acredita ser capaz de realizar na iteração e se compromete.

**Exemplifique sobre a habilidade na tomada de decisão e habilidade de solução de problemas confusos.**

Em deparo de um problema confuso e complexo, a equipe pode tomar a decisão de postergar aquela atividade para assim não perder a agilidade do projeto.

**Como funciona a auto-organização?**

É a própria equipe que deve se organizar para realizar o trabalho. É preciso procurar a todo instante o que mais pode ser melhorado no processo para que ele se adeque ainda mais ao ambiente. A auto organização tem benefícios técnicos, mas é muito importante para melhorar a colaboração. A equipe seleciona quanto trabalho acredita ser capaz de realizar na iteração e se compromete.

**Estória 6. | Grupo-2 | Bruna Kelly**

**Descreva os cinco valores que são a base para todo projeto realizado como parte da XP e quais seus conceitos.**

**Comunicação:** O cliente de um projeto de software tem um conjunto de problemas que deseja solucionar com o sistema em desenvolvimento e possui algumas idéias sobre que funcionalidades podem resolvê-los. Por sua vez, desenvolvedores possuem conhecimento sobre aspectos técnicos que influenciam a forma de solucionar o problema do cliente. Para que os desenvolvedores compreendam o que o cliente deseja e este último entenda os desafios técnicos que precisam ser vencidos, é preciso que haja comunicação entre as partes.

**Coragem:** Costuma-se dizer que a única constante em um projeto de software é a mudança. Clientes mudam de idéia com freqüência, mesmo quando fecham [contratos](http://www.desenvolvimentoagil.com.br/xp/praticas/contrato) nos quais, teoricamente, assumem o compromisso de não alterar o que está na especificação. Eles mudam porque aprendem durante o projeto e descobrem problemas mais prioritários a serem solucionados ou formas mais apropriadas de resolvê-los. Embora isso seja natural, gera uma preocupação para a equipe de desenvolvimento que, de tempos em tempos, precisa alterar partes do sistema que já estavam prontas, correndo o risco de se quebrar o que já vinha funcionando.

[XP](http://www.desenvolvimentoagil.com.br/xp) não tem uma solução mágica para eliminar esse risco. Ele existe em um projeto [XP](http://www.desenvolvimentoagil.com.br/xp), como existe em qualquer outro. O que muda é a forma de lidar com ele.

**Feedback:** Algumas pessoas seriam capazes de caminhar na beirada de um precipício com os olhos fechados, ou colocar a maior parte do seu dinheiro em um investimento com elevada chance de prejuízo sem acompanhá-lo de perto. Entretanto, a maioria das pessoas provavelmente manteria os olhos bem abertos em ambos os casos. Isso é particularmente verdade no caso de equipes trabalhando com [XP](http://www.desenvolvimentoagil.com.br/xp). Elas acreditam que projetos de software são iniciativas freqüentemente caras, arriscadas e com um histórico repleto de falhas, o que as leva a simples conclusão de que provavelmente o projeto em que estão envolvidas também enfrentará falhas e problemas, como é habitual na área de software.

Isso pode ser tratado de forma mais econômica e eficaz se as falhas forem detectadas rapidamente. Normalmente, quanto mais cedo descobrimos um problema, menos prejuízos ele pode causar e maiores são as chances de resolvê-lo de forma barata. Por isso, projetos [XP](http://www.desenvolvimentoagil.com.br/xp)estabelecem formas de encurtar ao máximo a defasagem de tempo entre o momento em que uma ação é executada e o seu resultado é observado.

**Respeito:** Respeito é um valor que dá sustentação a todos os demais. Membros de uma equipe só irão se preocupar em comunicar-se melhor, por exemplo, se se importarem uns com os outros. Respeito é o mais básico de todos os valores. Se ele não existir em um projeto, não há nada que possa salvá-lo. Saber ouvir, saber compreender e respeitar o ponto de vista do outro é essencial para que um projeto de software seja bem sucedido.

**Simplicidade:** O [XP](http://www.desenvolvimentoagil.com.br/xp) utiliza o conceito de simplicidade em inúmeros aspectos do projeto para assegurar que a equipe se concentre em fazer, primeiro, apenas aquilo que é claramente necessário e evite fazer o que poderia vir a ser necessário, mas ainda não se provou essencial.

**Estória 7. | Grupo-2 | Bruna Kelly**

**Explique resumidamente sobre as quatro atividades metodológicas do processo XP.**

Planejamento - Define o que é ou não necessário ser feito no projeto. A XP baseia-se em requisitos atuais para desenvolvimento de software, não em requisitos futuros.

Entregas Freqüentes - Baseiam-se no desenvolvimento de um software simples, e conforme os requisitos aparecem, há a atualização da versão do software. Cada versão entregue deve ter o menor tamanho possível, contendo os requisitos de maior valor para o negócio. É recomendado que as versões devem ser entregues a cada mês, ou no máximo a cada dois meses, aumentando a possibilidade de feedback rápido do cliente.

Metáfora - São as descrições de um software sem a utilização de termos técnicos com o objetivo de guiar o desenvolvimento do software com a maior transparência possível para o cliente.

Projeto simples - O software desenvolvido de acordo com a metodologia XP deve ser o mais simples possível e satisfazer os requisitos atuais, sem a preocupação de requisitos futuros. Eventuais requisitos futuros devem ser adicionados assim que eles realmente existirem.

Testes - A Extreme Programming (XP) prioriza a validação do projeto durante todo o processo de desenvolvimento. Os desenvolvedores implementam o software criando primeiramente os testes.