

Nome: Kerby Lovince
Matricula: 2011100019
Material: Engenharia Software I
Trabalho:

Pesquisar sobre a adoção dos métodos Scrum e eXtreme Programing:

- Pesquisar um estudo de caso ou relato sobre a adoção dos métodos Scrum e XP (eXtreme Programing)
- Fazer descrição sobre o estudo encontrado
- Apresentar os pontos que você achou mais importante/interessante sobre o relato

Obs: Deve constar o link das fontes (páginas) onde foi pesquisado o estudo \pagina3

Link: <http://www.periodicosibepes.org.br/index.php/reinfo/article/view/146/38>

De acordo com o texto posso dizer que a metodologia ágeis traz as vantagens para o desenvolvimento do software, com a metodologia ágeis a recomendação final final foi que o desenvolvimento de software deveria ser baseado em modelos incrementais, o que poderia evitar muitas das falhas reportadas.

As metodologias ágeis é que elas são adaptativas ao invés de serem preditivas. Com isso, elas se adaptam a novos fatores decorrentes do desenvolvimento do projeto, ao invés de procurar analisar previamente tudo o que pode acontecer no decorrer do desenvolvimento.

Enquanto as metodologias ágeis variam em termos de práticas e ênfases, elas compartilham algumas características, como desenvolvimento iterativo e incremental, comunicação e redução de produtos intermediários, como documentação extensiva. Desta forma existem maiores possibilidades de atender aos requisitos do cliente, que muitas vezes são mutáveis.,e existem vários modelos de metodologias. O modelo Cascata que é composto basicamente por atividades sequenciais de levantamento de requisitos, análise, projeto, implementação, teste, implantação e manutenção. Este modelo é derivado de outras engenharias tradicionais (Civil, Elétrica, Naval,...)

O modelo em Cascata dominou a forma de desenvolvimento de software até o início da década de 90, apesar das advertências dos pesquisadores da área e dos desenvolvedores,que identificaram os problemas gerados ao se adotar esta visão sequencial de tarefas. Por exemplo, Fred Brooks em seu famoso artigo “No Silver Bullet: Essence and Accidents of Software Engineering ”.

A maioria das metodologias ágeis nada possuem de novo [2]. O que as diferencia das metodologias tradicionais são o enfoque e os valores. A ideia das metodologias ágeis é o enfoque nas pessoas e não em processos ou algoritmos. Além disso, existe a preocupação de gastar menos tempo com documentação e mais com a implementação. Uma característica das metodologias ágeis é que elas são adaptativas ao invés de serem preditivas. Com isso, elas se adaptam a novos fatores decorrentes do desenvolvimento do projeto, ao invés de procurar analisar previamente tudo o que pode acontecer no decorrer do desenvolvimento.

Apresentar os pontos que você achou mais importante/interessante sobre o relato

O cliente então constantemente sugere novas características e informações aos desenvolvedores. Eventuais erros e não conformidades são rapidamente identificados e corrigidos nas próximas versões. Desta forma, a tendência é que o produto final esteja de acordo com as expectativas reais do cliente. É necessário coragem para implantar os três valores anteriores. Por exemplo, não são todas as pessoas que possuem facilidade de comunicação e têm bom relacionamento. A coragem também dá suporte à simplicidade, pois assim que a oportunidade de simplificar o software é percebida, a equipe pode experimentar. Além disso, é preciso coragem para obter feedback constante do cliente.

A prática do feedback constante significa que o programador terá informações constantes do código e do cliente. A informação do código é dada pelos testes constantes, que indicam os erros tanto individuais quanto do software integrado. Em relação ao cliente, o feedback constante significa que ele terá frequentemente, uma parte do software totalmente funcional para avaliar.

Projeto simples: o programa desenvolvido pelo método 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. Esta forma de raciocínio se opõe ao “implemente para hoje e projete para amanhã”.

Testes: a XP focaliza a validação do projeto durante todo o processo de desenvolvimento. Os programadores desenvolvem o software criando primeiramente os testes.