

Artigo

-> Introdução

- Utilização de casos de uso para levantamento de requisitos, por ter uma forma intencionalmente informal.
- Dificuldades de interpretação e de vinculações dos casos de uso por conta da informalidade
- Opções de que esse método é insuficiente
- O framework CASE ajudou muito mais do que não é uma baldeprato
- Surgimento do desenvolvimento ágil
- Histórias de usuários é uma boa forma de levantamento de requisitos.

-> Requisitos Leves (travessês de Histórias de Usuários

- Protocolos da História de Usuário

- Como a história de usuário é uma declaração informal de um requisito de usuário, fica mais fácil e rápido da equipe saber o sistema que tem que ser feito, ou seja, não é um requisito propriamente dito
 - A história é escrita em cartão
 - Textos de aceitação e especificam requisitos
 - Deve ser simples e legível por todos
- Como em...
- Eu quero...
- de modo que...

- Distinção da História de Usuário

- A conversa com o cliente irão interpretar um mesmo ponto da mesma forma
- Facilita o planejamento do projeto
- Dificuldade de priorização
- O levantamento de requisitos podendo ser feito em etapas é uma vantagem

- Histórias de Usuário não são Casos de Uso

- Casos de Uso é uma descrição generalizada de um conjunto de interações
- Diferença de abrangência das dois como a amplitude, ou seja, as histórias de usuário são apenas um cenário de um caso de uso principal.
- Propósito do caso de uso é documentar um acordo entre o cliente e a equipe de desenvolvimento
- Propósito da história de usuário é facilitar a liderança e planejamento da interação

- Histórias de Usuário não são Requisitos

- Padrão de levantamento de requisitos IEEE 830 "O sistema deve[...]"
- Requisitos sugerem o que deve ser feito e histórias de usuário focam no objetivos
- Diferença entre histórias de usuário e IEEE 830 é que o custo do requisito, ou seja, estimativa de tempo de implementação.

- Envolvendo o Cliente no Processo de Desenvolvimento

- O cliente está disponível para ajudar a equipe de desenvolvimento
- O usuário do negócio irá ajudar a ~~equipe~~ equipe de desenvolvimento a priorizar as histórias de usuários
- Com as entregas das histórias faz com que o cliente se sinta como parte do negócio e que sua colaboração contribuiu para a entrega final do projeto.

- Testes de Aceitação

- Confirmar / Testar deve assegurar que os objetivos da história sejam cumpridos.
- Geralmente são escritos no verso do cartão das histórias de usuário.
- Representa algum resultado esperado a partir do sistema

→ Estudo de Caso

- Descrição do Projeto

- O objetivo é compreender o que o sistema precisa fazer, bem como analisar o suficiente para que possa ser estimado.
- Fato entrevistados
- Os requisitos de negócios não se equivalem os de requisitos
- Flexibilidade para mudanças

- Aplicações do Caso de Uso

- A eficácia de um caso de uso depende da capacidade do autor para escrevê-los.
- O caso de uso contém informações específicas ~~que~~ sobre os passos que o usuário deve seguir para usar com sucesso a nova funcionalidade do sistema.

- Aplicação da História de Usuário

- A ideia é que seja apenas uma pessoa a revisar as histórias e priorizá-las para repassar a equipe de desenvolvimento.
- Estimativa das histórias para o Product Backlog

- Testes de Aceitação de Estudo de Caso

- O teste garante que a iteração atende os devidos critérios de aceitação
- Seta da caixa preta, onde se vê a funcionalidade em específico

- Comparação da História de Usuário com o Caso de Uso

- Casos de Uso são mais representativos do que histórias de usuário
- Principal diferença é que as histórias de usuário têm o objetivo de capturar a visão que o usuário tem sobre o sistema.

- Otimizando a História de Usuário

- Independência entre as histórias
- Devem gerar valor ao cliente
- Devem ser negociáveis
- Gerar histórias de menor tamanho possível

Vídeo

- Hoje está muito encolerto o que realmente são métodos ágeis
- É decadência dos métodos ágeis
- Cunda de métodos e não de processos e produtos
- Recomendando o XP
- Não se vende ágil
- Ainda existe como um zumbi
- "Não existe metodologia ágil existe métodos e práticas adotadas por programadores"
- Ágil significa:
 - 1- Entender onde você está
 - 2- ~~Entender~~ dar um pequeno passo em direção ao que quer ir
 - 3- Avaliar o que aconteceu
 - 4- Manter isso em loop (aproximação minha)
- Não querer comparar Engenheiros de atributos físicos para software
- É fácil mentir com números
- Ninguém gosta de estimar
- Não se importa mais com o não desperdício
- Riscos existe e são caros
- Ser ágil e ser adulto
- Não se esconder atrás de metodologias