

Programa de Pós-Graduação em Engenharia de Teleinformática
TIP8415 – Engenharia de Software
Métodos Ágeis e Documentação de Software
Artur Rodrigues Rocha Neto – 431951 – Abril/2019

1. De maneira geral, qual é a grande diferença entre a documentação em um método tradicional e em um método ágil?

A documentação possui papéis distintos em métodos tradicionais e ágeis. O volume de documentação na agilidade é menor, mas ela não é totalmente descartada. Nos métodos tradicionais, a documentação define aquilo que será desenvolvido, antes mesmo de começar o desenvolvimento. Já na agilidade, documentar faz parte do processo de desenvolvimento se isto trouxer valor.

2. Qual o fluxo de documentação comumente adotado em métodos tradicionais? E em métodos ágeis?

Nos métodos tradicionais, o cliente expressa seus desejos que gostaria de alcançar com o software a um analista que, por sua vez, trabalha uma lista de requisitos. Esses requisitos são então modelados por um arquiteto de software que gera o desenho do produto. Por fim, os desenvolvedores executam a implementação.

Já na agilidade, o processo de documentação acontece ao longo do desenvolvimento, sempre respeitando uma regra: “documente se o valor do documento for maior que o seu custo de criação e manutenção, caso contrário descarte-o.”

3. Cite vantagens associadas à documentação em métodos tradicionais e em métodos ágeis.

A documentação formal representa uma espécie de “contrato” firmado entre cliente e fornecedor de software, ou seja, uma forma de resguardar ambas as partes daquilo que foi requisitado e desenvolvido. Os métodos tradicionais também se valem da documentação como forma de familiarizar novos membros de equipe agregados no meio de projetos, tornando os documentos verdadeiros registros históricos do processo. Portanto, a documentação serve para comunicar, definir e referenciar.

A agilidade faz uso de pequenas histórias de usuário, interação informal dos desenvolvedores e práticas para melhoria da qualidade do código como arcabouço de documentação, valorizando software funcional sobre documentação vasta.

4. Cite desvantagens associadas à documentação em métodos tradicionais e em métodos ágeis.

A linguagem escrita e até mesmo a de modelagem não são sempre a melhor forma de comunicar e definir ideias. Além disso, requisitos mudam com facilidade, tornando inúteis documentos anteriormente criados. Esses aspectos tornam a documentação tradicional fraca em difusão e sensível a mudanças.

Os métodos ágeis recomendam o uso da verbalização para a transmissão dos requisitos, normalmente guardados na forma de pequenas histórias do usuário. Entretanto, essas práticas são difíceis de se reproduzir em grandes equipes, agravando-se em configurações remotas. Sem um registro histórico formal, a capacidade de acolher novos membros ao projeto fica comprometida.

5. Quais técnicas são incentivadas nas metodologias ágeis para reduzir a documentação formal?

Desenho simplificado de projeto, Desenvolvimento Guiado por Testes (TDD), refatorização e documentação de código.

6. Partindo de volumes pequenos de documentação, como os métodos ágeis respondem a dúvidas sobre requisitos?

Os métodos ágeis incentivam o desenvolvimento de pequenos incrementos, simples de se entender e reproduzir com a pequenas conversas entre os membros da equipe. Caso os questionamentos

persistam, é necessária a presença de uma figura que tenha maior conhecimento do domínio do problema. No SCRUM, esse indivíduo é o Product Owner, mas este pode ser também o próprio cliente, caso esteja disponível.

7. Como as metodologias ágeis enxergam a presença do cliente durante o desenvolvimento? Comente sobre as implicações desse relacionamento.

A agilidade prioriza colaboração com o cliente ao invés da negociação de contratos. Assim, esperasse uma menor quantidade de documentos para resguardar as partes. Entretanto, é importante apontar que a proximidade do cliente não pode ser garantida em todo projeto, trazendo incerteza em momentos de decisão.

8. Quais componentes compõem uma história de usuário? Exemplifique.

- **Funcionalidade:** O nome ou breve descrição do que deve ser implementado
- **Como/quem:** quem irá utilizar a funcionalidade
- **Quero:** a necessidade a ser atendida
- **Para:** objetivo ou resultado após a implementação

Exemplo:

Funcionalidade: Geração de relatórios

Como/quem: administrador do sistema

Quero: sumarização de entrada e saída por período de tempo

Para: ajudar na gestão de recursos