

Engenharia de Software 2


Revisão e conceitos gerais

Professor: Bruno Henrique
de Paula Ferreira

Fatec

Araras
Antonio Brambilla

Roteiro

- 
- Mitos
 - Conceitos
 - Questionário

Eternos mitos

Pressman classifica os mitos em três grupos:

- Administrativos
- Cliente
- Profissional

Mitos - Administrativos

A existência de um manual de procedimentos e padrões é suficiente para a equipe produzir com qualidade.

- Na verdade, deve-se questionar se o manual é realmente usado, se ele é completo e atualizado. Deve-se trabalhar com processos que possam ser gerenciáveis e otimizados, ou seja, sempre que a equipe identificar falhas no processo, deve haver outro processo para modificá-lo.

A empresa deve produzir com qualidade, pois tem ferramentas e computadores de última geração.

- Ferramentas e computadores de boa qualidade são condições necessárias, mas não suficientes.

Mitos - Administrativos

Se o projeto estiver atrasado, sempre é possível adicionar mais programadores para cumprir o cronograma.

O desenvolvimento de software é uma tarefa altamente complexa. Adicionar mais pessoas sem que haja um planejamento prévio pode causar mais atrasos.

Um bom gerente pode gerenciar qualquer projeto.

Gerenciar não é fazer, mesmo que o gerente seja competente, se não houver boa comunicação com a equipe, competência técnica e um processo de trabalho previsível, ele pouco poderá fazer para obter um produto com a qualidade desejada.

Mitos - Cliente

Os requisitos mudam com frequência, mas sempre é possível acomodá-los, pois o software é flexível.

- O *código* é fácil de mudar – basta usar um editor. Mas mudar o código sem introduzir erros é uma tarefa bastante improvável, especialmente em empresas com baixa maturidade de processo.

Eu sei do que preciso.

- O cliente não sabe do que precisa. É necessário que os analistas entendam que os clientes (a não ser que sejam técnicos especializados) raramente sabem do que realmente precisam e têm grande dificuldade para descrever e até mesmo para lembrar-se de suas necessidades.

Mitos - Profissional

Assim que o programa for colocado em operação, nosso trabalho terminou.

- Ainda haverá muito esforço a ser despendido depois da instalação do sistema, por causa de erros dos mais diversos tipos.

Se eu esquecer alguma coisa, posso arrumar depois.

- Quanto mais o processo de desenvolvimento avança, mais caras ficam as modificações em termos de tempo e dinheiro.

A única entrega importante em um projeto de software é o software funcionando.

- Se os usuários não conseguirem utilizar o sistema ou se os dados não forem corretamente importados, pouco valor ele terá. É necessário realizar testes de operação, importar dados, treinar usuários, definir procedimentos operacionais...

Engenharia de Software

- Engenharia de software é uma profissão dedicada a projetar, implementar e modificar software, de forma que ele seja de alta qualidade, a um custo razoável, manutenível e rápido de construir (Laplante, 2007).
- Engenharia de software é a aplicação de abordagens sistemáticas, disciplinadas e quantificáveis ao desenvolvimento, operação e manutenção de software, além do estudo dessas abordagens (IEEE Computer Society, 2004)

Papeis na Engenharia de Software

- O **engenheiro de software** escolhe e especifica os processos de planejamento, gerência e produção a serem implementados. Ele acompanha e avalia o desenvolvimento de todos os projetos da empresa para verificar se o processo estabelecido é executado de forma eficiente e efetiva. Caso sejam necessárias mudanças no processo estabelecido, ele as identifica e realiza, garantindo que a equipe adote tais mudanças. Ele reavalia o processo continuamente.

Papeis na Engenharia de Software

O **gerente de projeto** cuida de um projeto específico, garantindo que os prazos e orçamento sejam cumpridos. Ele segue as práticas definidas no processo de engenharia. É responsável por verificar a aplicação do processo pelos desenvolvedores e, se necessário, reporta-se ao engenheiro de software para sugerir melhorias.

O **analista** é um desenvolvedor responsável pela compreensão do problema relacionado ao sistema que se deve desenvolver, ou seja, pelo levantamento dos requisitos e sua efetiva modelagem. O analista deve, portanto, descobrir o que o cliente precisa. Por exemplo, controlar suas vendas, comissões, produtos etc.

Papeis na Engenharia de Software

O **designer** deve levar em conta as especificações do analista e propor a melhor tecnologia para produzir um sistema executável para elas. Deve, então, apresentar uma solução para as necessidades do cliente. Por exemplo, propor uma solução baseada em web, com um banco de dados centralizado acessível por dispositivos móveis etc.

O **programador** vai construir a solução física a partir das especificações do designer. É ele quem gera o produto final, e deve conhecer profundamente a linguagem e o ambiente de programação, bem como as bibliotecas que for usar, além de ter algum conhecimento sobre teste e depuração de software.

Tipos de Software

Software básico: são os compiladores, drivers e componentes do sistema operacional.

Software de tempo real: são os sistemas que monitoram, analisam e controlam eventos do mundo real.

Software comercial: são os sistemas aplicados nas empresas, como controle de estoque, vendas etc. Tais sistemas usualmente acessam bancos de dados. São também conhecidos como sistemas de informação.

Software científico e de engenharia: são os sistemas que utilizam intenso processamento de números.

Software embutido ou embarcado: são os sistemas de software presentes em celulares, eletrodomésticos, automóveis etc. Normalmente, tais sistemas precisam trabalhar sob severas restrições de espaço, tempo de processamento e gasto de energia.

Tipos de Software

Software pessoal: são os sistemas usados por pessoas no dia a dia, como processadores de texto, planilhas etc.

Jogos: embora existam alguns jogos cujo processamento não é muito complexo, existem também aqueles que exigem o máximo dos computadores em função da qualidade de gráficos e da necessidade de reação em tempo real. Apesar disso, todas as categorias de jogos têm características intrínsecas que extrapolam o domínio da engenharia de software.

Inteligência artificial: são os sistemas especialistas, redes neurais e sistemas capazes de alguma forma de aprendizado. Além de serem sistemas independentes, com um tipo de processo de construção próprio, podem também ser embutidos em outros sistemas.

Processos

Um processo de engenharia de software é formado por um conjunto de passos de processo parcialmente ordenados, relacionados com artefatos, pessoas, recursos, estruturas organizacionais e restrições, tendo como objetivo produzir e manter os produtos de software finais requeridos.

Um processo, então, pode ser entendido como um conjunto de atividades:

- a) Interdependentes.
- b) Com responsáveis.
- c) Com entradas e saídas definidas.

Projeto x Processo

- **Projeto** é algo que ocorre em um tempo determinado. Consiste na execução concreta de um conjunto de atividades que visam à criação de um produto específico.
- **Processo** é um conjunto de regras que definem como um projeto deve ser executado. No jargão da orientação a objetos, o projeto pode ser considerado uma instância de um processo.

Modelo de processo

É um conjunto de regras mais abstratas que especificam a forma geral de processos. Um modelo de processo apresenta uma filosofia, uma forma geral de comportamento com base na qual processos específicos podem ser definidos.

Também conhecido como ciclo de vida.

Fases

- Embora a nomenclatura possa variar de um modelo de processo para outro, usualmente se considera que a primeira grande divisão de um processo é a fase. Uma fase é um período de tempo no qual determinadas atividades são realizadas.
- Alguns processos, como o Modelo Cascata e suas variantes, têm fases sequenciais. Outros modelos podem ter fases cíclicas, ou seja, o desenvolvimento passa repetidamente de uma fase para outra formando um ciclo repetitivo de fases até a finalização do projeto. (Espiral e prototipação evolucionária)

Atividades ou tarefas e artefatos

- Toda atividade / tarefa tem um objetivo principal estabelecido e visa criar ou produzir uma mudança de estado visível em um ou mais artefatos durante a execução de um projeto.
- Artefatos são quaisquer documentos que puderem ser produzidos durante um projeto de desenvolvimento de software, incluindo diagramas, programas, documentos de texto, desenhos, contratos, projetos, planos etc.



Como o cliente
explicou...



Como o líder de projeto
entendeu...



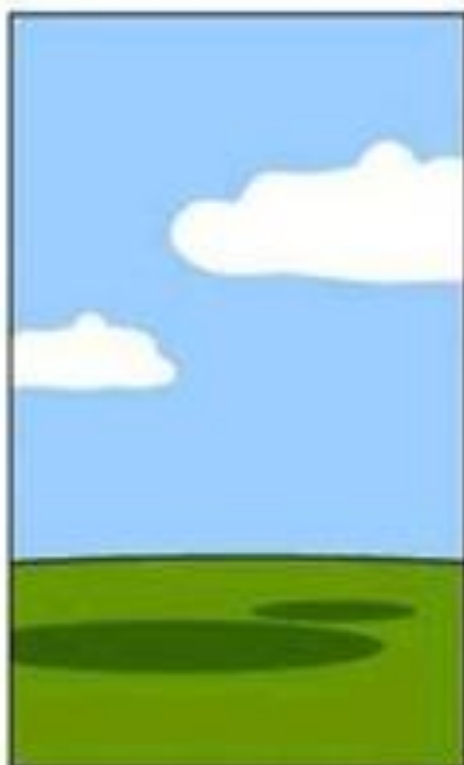
Como o analista
projetou...



Como o programador
construiu...



Como o Consultor de
Negócios descreveu...



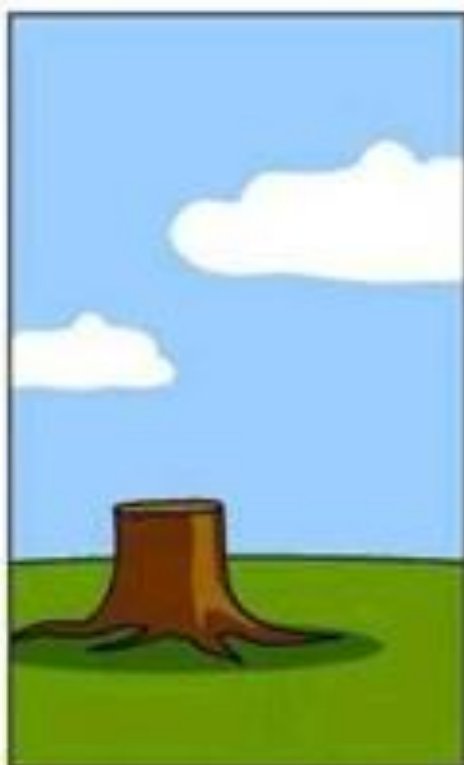
Como o projeto foi documentado...



Que funcionalidades foram instaladas...



Como o cliente foi cobrado...



Como foi mantido...



O que o cliente realmente queria...



Questionário