

Aula 01 - Engenharia de Requisitos

Software

Software é uma tecnologia muito importante para a nossa sociedade , e está presente em grande parte das nossas atividades do dia a dia. O processo de construção de software deve ser cada vez mais parecido com um processo de engenharia.

O software precisa ter qualidade, precisa fazer tudo aquilo que o cliente ou usuário necessitam, precisa ser confiável, auditável e seguro , também precisa ser feito dentro do prazo e dentro do custo.

Para se conseguir construir software com essas qualidades, é necessário disciplina no desenvolvimento de software.

Software pode ser definido como:

1. Instruções (programas de computador) que, quando executadas, provêm as características, funcionalidades e desempenho desejados
2. Estruturas de dados que permitem aos programas, manipular informação de forma eficiente.

O Software é desenvolvido, ele não é fabricado, além de que ele não se desgasta. Apesar da indústria estar a caminho do desenvolvimento baseado em componentes, a maior parte do software é feito sob medida.

O software legado foi desenvolvido décadas atrás e tem sido continuamente modificado para atender às necessidades do negócio ou atualizações de plataforma computacional. O software legado é atualizado, mas não é substituído.

Frequentemente, o software legado foi construído sem muita preocupação com qualidade ou documentação, além de ter sido desenvolvido em linguagens de programação que não são mais muito usadas. Isto faz com que sua adaptação seja cada vez mais custosa.

Engenharia de software

É o estabelecimento e uso de sólidos princípios de engenharia para que se possa obter, de forma econômica, um software que seja confiável e que funcione eficientemente em máquinas reais. A engenharia de software abrange:

- Ferramentas: fornecem apoio automatizado ou semi-automatizado para o processo e os métodos
- Métodos: fornecem a técnica de “como fazer” para construir software.
- Processo: Alicerce da engenharia de software que mantém unidas as camadas de tecnologia.
- Foco na qualidade: compromisso organizacional com a qualidade.

Atividades de arcabouço

- Comunicação (eng. Requisitos)
- Planejamento (Plano, cronograma, riscos, recursos, tarefas)
- Modelagem
- Construção
- Implantação

Atividades “Guarda-chuva”

- Acompanhamento e controle de projeto de software
- Gestão de risco
- Garantia da qualidade de software
- Revisões técnicas e formais
- Medição
- Gestão de configuração de software
- Gestão de reusabilidade
- Preparação e produção do produto de trabalho (documentos, modelos, registros, formulários e listas)

Processo de software

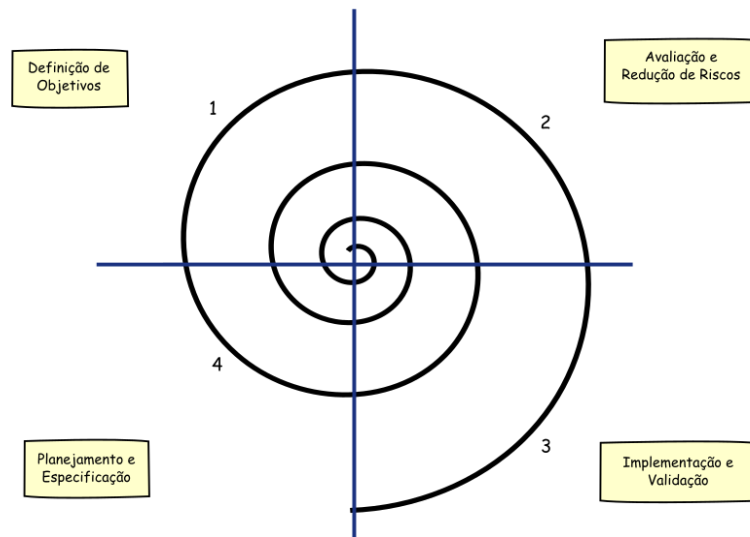
O processo de desenvolvimento de software pode ser definido como uma sequência de etapas para a construção do software. Cada etapa tem objetivos bem definidos e gera um conjunto de artefatos.

O processo de desenvolvimento de software está inserido no ciclo de vida do software. Um modelo de ciclo de vida, diz quais são as atividades que devem ser realizadas durante o desenvolvimento do software e durante o seu uso.

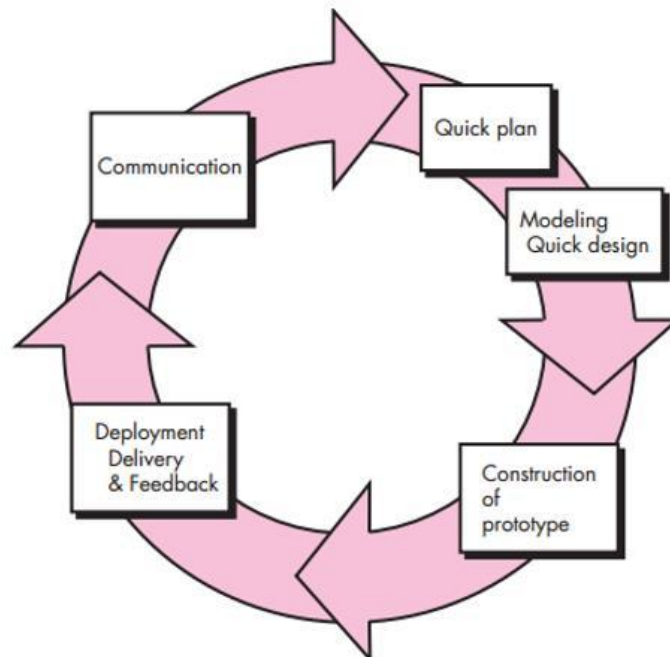
Modelos de ciclo de vida

- Modelo Codificar e Corrigir
 - Vantagens:
 - Não se gasta tempo com documentação, planejamento ou projeto;
 - É fácil visualizar o progresso;
 - Não exige treinamento especial no processo;
 - Desvantagens:
 - Difícil avaliar a qualidade;
 - Qualquer mudança arquitetural vai impactar em começar tudo do início.
- Modelo tradicional (Cascata)
 - Vantagens:
 - Existência de fases bem definidas ajuda a detectar erros cedo;
 - O modelo procura promover a estabilidade dos requisitos;

- Funciona bem com requisitos bem conhecidos e estáveis;
 - Vantajoso quando a preocupação da qualidade é fundamental;
 - Adequado para equipes inexperientes.
- Desvantagens:
 - Não produz resultados práticos até a fase de codificação;
 - Difícil estabelecer requisitos completos antes da codificação;
 - Não há flexibilidade nos requisitos;
- Modelo espiral



- Modelo de prototipação



- Modelo Iterativo e Incremental

