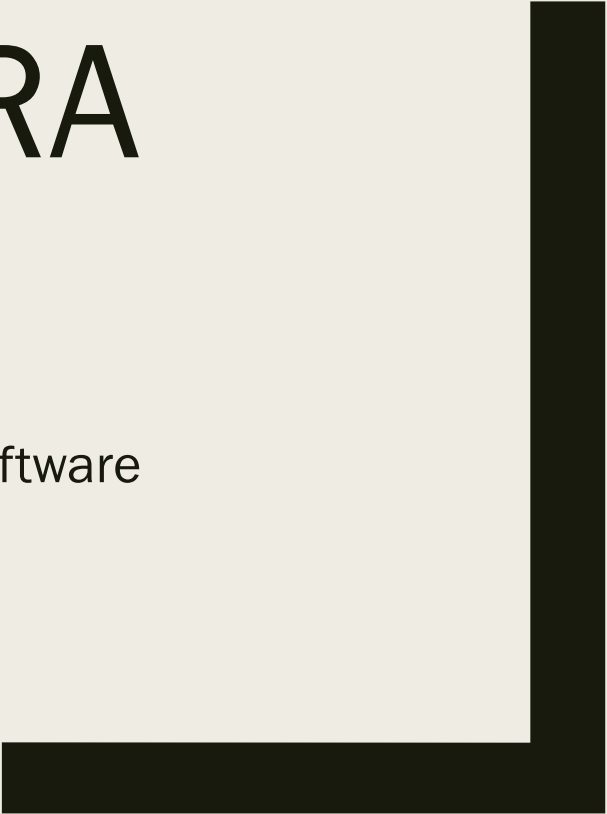




REVISÃO PARA PROVA

Disciplina: Processo de Produção de Software
Prof^a. Adriana Herden



DEFINA SISTEMA

[cap 2 – Sommerville 8a. edição]

- *Um sistema é o conjunto intencional de componentes inter-relacionados que funcionam juntos para atingir certo objetivo. Uma característica de todos os sistemas é que as propriedades e o comportamento dos componentes do sistema estão fortemente interligados. O funcionamento com sucesso de cada componente depende do funcionamento dos outros componentes.*

DEFINA SISTEMA

[cap 2 – Sommerville 8a. edição]

- *Sistemas que incluem software se concentram em 2 categorias:*
 - *Sistemas técnicos baseados em computadores – incluem hardware e software, mas não incluem procedimentos e processos. Os indivíduos e as organizações usam sistemas técnicos para algum propósito, mas o conhecimento desse propósito não é parte do sistema. ex. Processador de texto.*
 - *Sistemas sociotécnicos – incluem um ou mais sistemas técnicos, mas, decisivamente, incluem também conhecimento de como o sistema deve ser usado para alcançar um objetivo maior. Isso significa que esses sistemas tem processos operacionais definidos, incluem pessoas como partes inerentes do sistema, são regidos pelas políticas e regras organizacionais e podem ser afetados por restrições externas, com leis e políticas regulamentadoras nacionais.*

Definição de SOFTWARE

[cap 1 e 2 – Pressman e Sommerville]

- Softwares são programas de computador e documentação associada. Produtos de software podem ser desenvolvidos para um cliente específico ou para o mercado em geral.
 - VERDADE
- Softwares abrangem programas executáveis em um computador de qualquer porte ou arquitetura, conteúdos, informações.
 - VERDADE
- Software não é apenas um programa, ele inclui também a documentação. Os atributos principais de um produto de software são manutenibilidade, confiança, proteção, eficiência e aceitabilidade.
 - VERDADE

ENGENHARIA DE SOFTWARE

[cap 1 e 2 – Pressman e Sommerville]

- Quais são os principais desafios da engenharia de software?
 - *Lidar com o aumento da diversidade, demandas pela diminuição do tempo para entrega e desenvolvimento de software confiável.*
- Engenharia de software abrange um **processo**, um conjunto de **métodos** (práticas) e um conjunto de **ferramentas** que possibilitam aos profissionais desenvolverem software de qualidade.
 - *VERDADE*
- Engenharia de software é uma disciplina de engenharia que se preocupa com todos os aspectos de produção de software.
 - *VERDADE*

ENGENHARIA DE SOFTWARE

[SWEBOK – Guide to the Software Engineering Body of Knowledge]

- Quais são as áreas de conhecimento da engenharia de software segundo SWEBOK?
 - *TODAS OPÇÕES*
 - *[Requisitos de Software, Projeto de Software, Construção de Software, Teste de Software, Manutenção de Software, Gerência de Configuração de Software, Gerência de Engenharia de Software, Ferramentas e Métodos da Engenharia de Software, Qualidade de Software]*

Quais são os componentes da Engenharia de Software?

[cap 1 – Pressman]

- *Processo (Procedimento)*
 - *Leva à produção do software*
- *Métodos*
 - *Oferecem a técnica de “como fazer”*
- *Ferramentas*
 - *Oferecem apoio automatizado (ou semi), para o processo e para os métodos.*



MODELOS DE PROCESSO DE SOFTWARE

[cap 2 e 3 – Pressman e Sommerville]

■ Defina Processo de Software

- *É um conjunto de atividades e resultados associados que produz um produto de software. (Sommerville, 2007);*
- *É um arcabouço para as tarefas que são necessárias para construir softwares de alta qualidade. Define a abordagem que é adotada quando o software é elaborado.(Pressman, 2006).*
- *Um processo define “quem”, está fazendo “o que”, “quando” e “como”, para atingir determinada meta que é a construção do software. (Jacobson, Booch e Rumbaugh, 1999);*

MODELOS DE PROCESSO DE SOFTWARE

[cap 2 e 3– Pressman e Sommerville]

- Qual a diferença entre um modelo de processo de software e ciclo de vida do software?
 - *Modelo de processo de software é uma representação abstrata de um processo de software. E um projeto de desenvolvimento concretiza esta abstração.*
 - *Ciclo de Vida é definido como uma sequência de Fases que buscam uma meta específica no tempo de existência do software.*
- A modelagem de processo de software descreve a criação de modelos do processo de desenvolvimento de software, referindo-se à definição de processos como modelos.
- Quais são as atividades fundamentais de um modelo de processo de software?

Atividades Fundamentais de Processo de Software

[cap 2 e 3– Sommerville]

- Especificação
 - *Definição do software e suas restrições*
- Desenvolvimento
 - *Programação (projeto e implementação)*
- Validação
 - *Verificação da qualidade com cliente*
- Evolução
 - *Modificação conforme requisitos do cliente/mercado*

Arcabouço de Processo Genérico de Software

[cap 2 e 3 – Pressman]

- Comunicação
 - *Colaboração com o cliente*
 - levantamento de requisitos
- Planejamento
 - *Plano de trabalho*
 - tarefas, riscos, recursos, produtos e cronograma
- Modelagem
 - *Criação de Modelos (Análise e Projeto)*
- Construção
 - *Geração de código e testes*
- Implantação
 - *Entrega do software para o cliente*

MODELOS DE PROCESSO DE SOFTWARE

[cap 2 e 3– Pressman e Sommerville]

- Modelo Espiral é usado quando os requisitos são bem compreendidos e pouco provavelmente venham a ser radicalmente alterados durante o desenvolvimento do sistema.
 - *FALSO. MODELO CASCATA*

- Modelo Cascata é centrado em torno de um conjunto de sprints, que são períodos determinados de tempo, quando um incremento de sistema é desenvolvido. O planejamento é baseado na priorização de um backlog de trabalho e na seleção das tarefas mais importantes para um sprint.
 - *FALSO. FRAMEWORK DE PROJETOS - SCRUM*

MODELOS DE PROCESSO DE SOFTWARE

[cap 2 e 3 – Pressman e Sommerville]

- No Scrum existem três fases. A primeira é uma fase de planejamento geral, em que se estabelecem os objetivos gerais do projeto e da arquitetura do software. Em seguida, ocorre uma série de ciclos de sprint, sendo que cada ciclo desenvolve um incremento do sistema. Finalmente, a última fase do projeto encerra o projeto e avalia as lições aprendidas.
 - *VERDADE.*
- Prototipação é um framework de processo de software dirigido a riscos. Após a avaliação dos riscos, algum desenvolvimento é efetivado, seguido por uma atividade de planejamento para a próxima fase do processo.
 - *FALSO. MODELO ESPIRAL*
- Processo Unificado é um modelo de processo de software iterativo e incremental, dirigido por casos de uso e centrado na arquitetura de software.
 - *VERDADE*

MODELOS DE PROCESSO DE SOFTWARE

[cap 2 e 3 – Pressman e Sommerville]

Classifique os modelos de processo de software quanto a execução do seu ciclo de vida:

- Modelo Cascata – **sequencial linear**
- Prototipação – **evolucionário (prototipação)**
- Modelo Espiral – **evolucionário**
- Metodos Ágeis (Scrum) – **Iterativo e Incremental**
- Processo Unificado – **Iterativo e Incremental**
- WebML – **Iterativo e Incremental**

MODELOS DE PROCESSO DE SOFTWARE

[cap 2 e 3 – Pressman e Sommerville]

- Quais são as principais diferenças entre métodos tradicionais e métodos ágeis?
 - *Acomodação a mudança em requisitos (iteração do processo)*
 - *Entrega incremental*

- Quais são as principais características dos métodos de desenvolvimento para web?
 - *Criação de modelos de hipertexto por visão de usuário (visões do site)*
 - *Sistemas Web são intensamente voltados para redes, guiados por conteúdo e evoluem continuamente.*

Ferramentas

- Indique uma ferramenta para Gerenciamento de Requisitos:
 - *EasyRM, Rational RequisitePro, RTM*
- Indique uma ferramenta para Gerenciamento de Projetos:
 - *MsProject, WBS tool*
- Indique uma ferramenta para Modelagem de Processo de Software:
 - *EPF Composer*
- Indique uma ferramenta para Prototipação de Interface - (Web e Mobile):
 - *PowerDesigner*

Ferramentas

- Indique uma ferramenta para Diagramação de Análise e Projeto Orientados a Objetos (baseado em UML):
 - *Visual Paradigm*
- Indique uma ferramenta para Testes Automatizados (pode ser em Java):
 - *Junit, TestWorks*
- Indique uma ferramenta para Implementação de Software (linguagem de programação é livre):
 - *VisualStudio, Eclipse*

Referências

- **PRESSMAN**, R. S. **Engenharia de software: uma abordagem profissional**. 7.ed. Porto Alegre: AMGH, 2011.
 - Cap 1 – Engenharia de Software
 - Cap 2 – Modelos de Processo
 - Cap 3 – Desenvolvimento Ágil
- **SOMMERVILLE**, I. **Engenharia de software**. 9.ed. São Paulo: Pearson Prentice Hall, 2011.
 - Cap 1 – Introdução
 - Cap 2 – Processo de Software
 - Cap 3 – Desenvolvimento Ágil de Software