

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PIAUÍ
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS – CSHNB
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO
DISCIPLINA DE ANÁLISE E PROJETO DE SISTEMAS

Atividade de Fixação I

João dos Santos Neto

1. Qual das seguintes atividades não está relacionada à análise de sistemas?

- a) Projeto de interface.
- b) Estudo de viabilidade.
- c) Implementação de código.
- d) Levantamento de requisitos.
- e) Análise de riscos.

Resposta: c) Implementação de código.

2. Qual dos seguintes métodos é usado para identificar os requisitos do sistema?

- a) Entrevistas.
- b) Codificação.
- c) Testes de unidade.
- d) Revisões de código.
- e) Desenvolvimento ágil.

Resposta: a) Entrevistas.

3. Qual das seguintes abordagens é mais adequada para lidar com requisitos em constante mudança?

- a) Modelagem de processos de negócios.
- b) Desenvolvimento em cascata.
- c) Programação estruturada.
- d) Desenvolvimento baseado em componentes.
- e) Desenvolvimento ágil.

Resposta: e) Desenvolvimento ágil.

4. Qual das seguintes opções descreve corretamente o Processo Unificado de Software?

- a) Um processo de desenvolvimento de software baseado em metodologia ágil.
- b) Um processo de desenvolvimento de software baseado em prototipagem.
- c) Um processo de desenvolvimento de software iterativo e incremental.
- d) Um processo de desenvolvimento de software baseado em modelo em cascata.
- e) Um processo de desenvolvimento de software baseado em testes.

Resposta: c) Um processo de desenvolvimento de software iterativo e incremental.

5. Quais são as fases principais do Processo Unificado de Software?

- a) Análise e Testes.
- b) Planejamento e Manutenção.
- c) Implementação e Documentação.
- d) Iniciação e Encerramento.
- e) Concepção, Elaboração, Construção e Transição.

Resposta: e) Concepção, Elaboração, Construção e Transição.

6. O que é uma iteração no contexto do Processo Unificado de Software?

- a) Uma reunião entre a equipe de desenvolvimento e os usuários finais.
- b) Um marco de progresso durante a implementação do software.
- c) Uma versão funcional do software pronta para lançamento.
- d) Um ciclo curto de trabalho que produz um incremento de software executável.
- e) Uma etapa final de revisão antes da entrega do produto final.

Resposta: d) Um ciclo curto de trabalho que produz um incremento de software executável.

7. Qual é o principal benefício do Processo Unificado de Software em comparação com a abordagem em cascata tradicional?

- a) Maior ênfase na documentação extensiva.
- b) Foco na entrega antecipada de um produto desenvolvido.
- c) Redução dos custos de desenvolvimento de software.
- d) Melhoria na comunicação entre os membros da equipe de desenvolvimento.
- e) Maior controle sobre cada etapa do ciclo de vida do software.

Resposta: b) Foco na entrega antecipada de um produto desenvolvido.

8. O que é um artefato no contexto do Processo Unificado de Software?

- a) Uma ferramenta específica usada para desenvolver softwares.
- b) Uma representação visual de um componente de software.
- c) Um item tangível produzido durante o ciclo de vida do software.
- d) Uma etapa específica do processo de desenvolvimento de software.
- e) Um documento ou resultado intermediário produzido durante o processo.

Resposta: e) Um documento ou resultado intermediário produzido durante o processo.

9. Qual é a principal característica do Processo Unificado de Software em termos de riscos?

- a) Evitar riscos a todo custo durante o desenvolvimento do software.
- b) Identificar e mitigar riscos o mais cedo possível no projeto.
- c) Adotar uma abordagem reativa para lidar com os riscos durante o desenvolvimento.
- d) Transferir todos os riscos para uma equipe de gerenciamento separada.
- e) Ignorar os riscos e focar exclusivamente na entrega rápida do software.

Resposta: b) Identificar e mitigar riscos o mais cedo possível no projeto.

10. Qual é o principal objetivo da fase de Elaboração no Processo Unificado de Software?

- a) Finalizar a documentação do sistema.
- b) Implementar todas as funcionalidades principais do software.
- c) Refinar a arquitetura do sistema e reduzir incertezas.
- d) Conduzir testes extensivos de desempenho e segurança.
- e) Iniciar a transição do software para o ambiente de produção.

Resposta: c) Refinar a arquitetura do sistema e reduzir incertezas.

11. Qual é o objetivo da fase de Transição no Processo Unificado de Software?

- a) Realizar testes finais de aceitação com os usuários finais.
- b) Refatorar o código-fonte para melhorar o desempenho.
- c) Documentar todos os componentes do sistema de maneira abrangente.
- d) Preparar o software para lançamento e uso em um ambiente de produção.
- e) Iniciar o processo de desenvolvimento de novas funcionalidades para o sistema.

Resposta: d) Preparar o software para lançamento e uso em um ambiente de produção.

12. O que é o padrão de arquitetura Model-View-Controller (MVC) em relação ao Processo Unificado de Software?

- a) Uma abordagem para desenvolver interfaces de usuário amigáveis.
- b) Um método para realizar testes de unidade no código-fonte.
- c) Um padrão para estruturar o código-fonte de maneira organizada.
- d) Uma técnica para otimizar o desempenho do software durante a transição.
- e) Um modelo de planejamento para a implementação de novas funcionalidades.

Resposta: c) Um padrão para estruturar o código-fonte de maneira organizada.

13. Quais são os principais pilares da filosofia do Processo Unificado de Software?

- a) Planejamento e execução.
- b) Comunicação e colaboração.
- c) Testes e depuração.
- d) Documentação e manutenção.
- e) Análise e implementação.

Resposta: b) Comunicação e colaboração.

14. Qual é a importância da avaliação contínua no contexto do Processo Unificado de Software?

- a) Garantir a conformidade com os padrões de codificação estabelecidos.
- b) Identificar e corrigir todos os bugs antes do lançamento do software.
- c) Manter um registro detalhado de todas as iterações e marcos de progresso.
- d) Melhorar a comunicação entre os membros da equipe de desenvolvimento.
- e) Garantir que o software atenda aos requisitos do cliente de maneira eficaz.

Resposta: e) Garantir que o software atenda aos requisitos do cliente de maneira eficaz.

15. Quais são os principais desafios enfrentados durante a implementação do Processo Unificado de Software?

- a) Falta de documentação detalhada do sistema.
- b) Conflitos entre os membros da equipe de desenvolvimento.
- c) Excesso de ênfase em testes de desempenho.
- d) Dificuldade em gerenciar as expectativas dos usuários finais.
- e) Problemas relacionados à escalabilidade do software durante a transição.

Resposta: b) Conflitos entre os membros da equipe de desenvolvimento.

16. O que são requisitos funcionais em um software?

- a) Requisitos relacionados ao desempenho do sistema.
- b) Requisitos relacionados à segurança do sistema.
- c) Requisitos que descrevem as funcionalidades do software.
- d) Requisitos relacionados à documentação do código fonte.
- e) Requisitos relacionados à manutenção do software.

Resposta: c) Requisitos que descrevem as funcionalidades do software.

17. Qual é a importância da elicitación de requisitos de software?

- a) Definir o design visual do software.
- b) Garantir a segurança dos dados do software.
- c) Estimar o custo do desenvolvimento de software.
- d) Coletar informações das partes interessadas.
- e) Testar a funcionalidade do software.

Resposta: d) Coletar informações das partes interessadas.

18. O que são requisitos não funcionais de software?

- a) Requisitos que descrevem as funcionalidades do software.
- b) Requisitos relacionados à usabilidade e desempenho.
- c) Requisitos que definem a estrutura do código fonte.
- d) Requisitos relacionados à interface do usuário.
- e) Requisitos que definem o ciclo de vida do software.

Resposta: b) Requisitos relacionados à usabilidade e desempenho.

19. Qual é o objetivo da análise de requisitos de software?

- a) Documentar o código fonte do software.
- b) Identificar possíveis conflitos e inconsistências nos requisitos.
- c) Desenvolver a interface do usuário do software.
- d) Testar a segurança do software.
- e) Elaborar manuais de usuário.

Resposta: b) Identificar possíveis conflitos e inconsistências nos requisitos.

20. Por que a documentação de requisitos de software é essencial?

- a) Para acelerar o processo de desenvolvimento de software.
- b) Para estabelecer diretrizes de marketing do software.
- c) Para melhorar a comunicação entre equipes e partes interessadas.
- d) Para aumentar a complexidade do software.
- e) Para reduzir o número de funcionalidades do software.

Resposta: c) Para melhorar a comunicação entre equipes e partes interessadas.

21. Quais são os principais desafios na gestão de requisitos de software?

- a) Definição clara de metas e objetivos.
- b) Comunicação eficaz entre equipes e partes interessadas.
- c) Identificação de requisitos funcionais.
- d) Documentação de requisitos de hardware.
- e) Teste de requisitos não funcionais.

Resposta: a) Definição clara de metas e objetivos.

22. O que são requisitos de software verificáveis?

- a) Requisitos que podem ser rastreados e testados.
- b) Requisitos que não podem ser implementados.
- c) Requisitos relacionados à segurança do software.
- d) Requisitos que definem a estrutura do software.
- e) Requisitos que descrevem as interfaces do usuário.

Resposta: a) Requisitos que podem ser rastreados e testados.

23. Qual é a finalidade da validação de requisitos de software?

- a) Confirmar a funcionalidade do software.
- b) Verificar a compatibilidade do software com hardware externo.
- c) Testar a documentação do software.
- d) Garantir a conformidade com os padrões de codificação.
- e) Confirmar que os requisitos atendem às necessidades dos usuários.

Resposta: e) Confirmar que os requisitos atendem às necessidades dos usuários.

24. O que é o processo de validação de requisitos de software?

- a) Revisão dos requisitos durante o ciclo de vida do software.
- b) Teste das funcionalidades do software antes da implementação.
- c) Identificação de conflitos entre equipes de desenvolvimento.
- d) Monitoramento do desempenho do software em tempo real.
- e) Avaliação da documentação técnica do software.

Resposta: a) Revisão dos requisitos durante o ciclo de vida do software.