**Marque as respectivas respostas no gabarito abaixo. Cada questão vale 0,6 pontos.**

| **QUESTÃO:** | **Q1** | **Q2** | **Q3** | **Q4** | **Q5** | **Q6** | **Q7** | **Q8** | **Q9** | **Q10** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RESPOSTA**: | B | D | B | B | B | E | B | A | B | D |

1. **(Ano: 2010 Banca: BIO-RIO)**. No contexto da Engenharia de Software, segundo SUMMERVILLE, o processo de engenharia de requisitos inclui subsistemas de alto nível. Além da Viabilidade e Elicitação, compreende:
2. Análise e projeto;
3. Implantação e manutenção;
4. Implementação e testes;
5. Especificação e validação;
6. Projeto físico e projeto lógico.
7. **(Ano: 2014 Banca: FUNCAB).** Em relação aos modelos de desenvolvimento de software, aquele que prioriza a análise dos riscos envolvidos no desenvolvimento de cada parte do software é:
8. De Prototipação
9. Cascata.
10. Desenvolvimento Baseado em Componentes.
11. Espiral.
12. Incremental.
13. **(Ano: 2007 Banca: CESGRANRIO)**. O ciclo de vida clássico da engenharia de software tem sofrido críticas dos especialistas. Uma delas é o fato de o cliente ter de declarar todos os requisitos explicitamente na primeira parte do projeto, gerando insegurança. Uma técnica utilizada para minimizar esse problema é conhecida como:
14. Análise por ponto de função.
15. Prototipação.
16. Casos de teste.
17. Linha básica ou baseline.
18. Projeto de software.
19. **(Ano: 2005 Banca: FAPEU)**. A respeito de teste de software, é INCORRETO afirmar que:
20. Teste de regressão consiste na reaplicação de testes já aplicados, em função de posteriores alterações do software testado.
21. O plano de teste de validação (de requisitos) só pode ser elaborado após a etapa de implementação.
22. Teste de integração avalia a compatibilidade entre módulos de software previamente testados.
23. Teste exaustivo corresponde à exploração de todas as possibilidades de execução de um software.
24. **(Ano: 2024 Banca: Instituto Consulplan).** É o seu pior pesadelo. Um cliente entra em seu escritório, senta-se, olha diretamente nos seus olhos e diz: “Eu sei que você pensa que entendeu o que eu disse, mas o que você não entende é que aquilo que eu disse não era o que eu quis dizer”. Invariavelmente, isso acontece no final do projeto, após compromissos de prazos de entrega terem sido estabelecidos, reputações estarem em risco e muito dinheiro estar em jogo. Quem trabalhou na área de software e sistemas por mais do que alguns poucos anos já viveu esse pesadelo; mesmo assim, poucos aprenderam a livrar-se dele. Passamos por muitas dificuldades ao tentar extrair os requisitos de nossos clientes. Temos dificuldades para entender as informações obtidas. Normalmente, registramos os requisitos de forma desorganizada e investimos pouco tempo verificando aquilo que registramos. Deixamos que as mudanças nos controlem, em vez de estabelecermos mecanismos para controlar as mudanças. Em suma, não conseguimos estabelecer uma base sólida para o sistema ou software. Todos esses problemas são desafiadores. Quando combinados, o panorama é assustador até mesmo para os gerentes e profissionais mais experientes. Mas soluções existem. (PRESSMAN, 2021, p. 103. Adaptado.) Sobre a Engenharia de Requisitos, marque V para as afirmativas verdadeiras e F para as falsas.

( ) Recomenda-se iniciar a engenharia de requisitos desde o início de um projeto de software para garantir que o software atenda às necessidades do cliente.

( ) Durante a tarefa de elaboração na engenharia de requisitos, são identificados e analisados os cenários de usuário.

( ) A principal finalidade da negociação durante a engenharia de requisitos é eliminar completamente os requisitos conflitantes.

( ) A especificação de requisitos no contexto da engenharia de requisitos é um processo iterativo de captura e documentação dos requisitos.

( ) A rastreabilidade é importante na engenharia de requisitos para garantir que todos os requisitos estejam alinhados com as diretrizes de engenharia de software.

A sequência está correta em:

1. V, F, V, F, V.
2. V, V, F, V, V.
3. V, V, F, V, F.
4. V, F, V, F, F.
5. **(Banca: Fundação Getúlio Vargas - FGV)**. No desenvolvimento centrado no usuário, tanto o conceito de MVP (Minimum Viable Product) quanto o de protótipo são essenciais. Acerca desse assunto, analise as afirmativas a seguir.
6. O protótipo permite testar ideias de produto no mercado real, reduzindo riscos financeiros ao evitar investimentos pesados antes de comprovar o interesse dos usuários.
7. O MVP deve apresentar o maior número possível de funcionalidades, de modo a avaliar corretamente o seu valor para o usuário.
8. Na engenharia de requisitos, os protótipos são empregados nas etapas de especificação, enquanto o MVP é empregado na validação.

Está correto o que se afirma em

1. I, apenas.
2. II, apenas.
3. I e II, apenas.
4. II e III, apenas.
5. III, apenas.
6. **(Ano: 2006 Banca: CEBRASPE)**. Com relação a engenharia de software, julgue os seguintes itens.
7. Já se tem livros repletos de padrões e procedimentos para elaborar software; isso fornece ao desenvolvedor tudo o que ele precisa.
8. A engenharia de software vai resultar na criação de documentação volumosa e desnecessária que certamente nos atrasará.
9. Se eu decidir terceirizar um projeto de software, vou poder relaxar e deixar que aquela firma o elabore.
10. Até que eu esteja com o programa executando, não tenho como avaliar a sua qualidade.
11. Os requisitos de projeto mudam continuamente, mas as mudanças podem ser facilmente acomodadas porque o software é flexível.
12. Possuímos ferramentas de desenvolvimento de software que estão no estado-da-arte, afinal compramos os computadores mais novos do mercado.
13. O único produto de trabalho que pode ser entregue para um projeto de software bem-sucedido é o programa executável.
14. Se o planejamento atrasar, é possível adicionar mais programadores e ficar em dia (conceito horda mongólica).
15. O estabelecimento geral de objetivos é suficiente para iniciar a escrita de programas; os detalhes podem ser fornecidos posteriormente.

Acerca da classificação desses itens, é correto afirmar que:

1. I, III, IV e IX dizem respeito a mitos da gerência.
2. V e IX dizem respeito a mitos do cliente.
3. II, IV e VI dizem respeito a mitos do profissional.
4. VI, VII e VIII dizem respeito a mitos de gerência.
5. **(Ano: 2024 Banca: Instituto Consulplan).** Relacione adequadamente os tipos de testes de software às suas respectivas descrições.

| 1. Unitário | ( ) Validar se o software é aceitável para uso de acordo com os requisitos e as necessidades de negócios. |
| --- | --- |
| 2. Integração | ( ) Garantir que essas partes funcionem bem juntas como um sistema coeso. |
| 3. Funcional | ( ) Verificar se o código-fonte de cada unidade funciona conforme o esperado. |
| 4. Aceitação | ( ) Certificar-se de que o software execute as ações esperadas e forneça os resultados corretos. |
| 5. Desempenho | ( ) Medir como o sistema se comporta sob diferentes condições de carga, identificando gargalos de desempenho. |

A sequência está correta em:

1. 4, 2, 1, 3, 5.
2. 1, 3, 2, 5, 4.
3. 4, 5, 3, 1, 2.
4. 3, 4, 2, 5, 1.
5. **(Ano: 2005 Banca: FAPEU - Adaptado)**. O Tribunal solicita o uso de métodos de teste de software para que o Analista de Sistemas possa derivar os casos de teste que:

* Garantam que todos os caminhos independentes dentro de um módulo tenham sido exercitados pelo menos uma vez;
* Exercitem todas as decisões lógicas para valores falsos ou verdadeiros;
* Executem todos os laços (loops) em suas fronteiras e dentro de seus limites operacionais;
* Exercitem as estruturas de dados internas para garantir sua validade.

Para atender a todos os requisitos acima, o Analista de Sistemas usa a abordagem de teste de

1. Particionamento de equivalência.
2. Caixa branca.
3. Caixa preta.
4. Implantação.
5. Caminho básico.
6. **(Ano: 2022 Banca: AMEOSC)**. "Existem muitos processos de software diferentes, mas todos devem incluir quatro atividades fundamentais para a engenharia de software: Especificação de software, Projeto e implementação de software, Validação de software, Evolução de software. De alguma forma, essas atividades fazem parte de todos os processos de software." *Sommerville, Ian. Engenharia de software (p. 18). Edição do Kindle.*

Relacione a Coluna 1 que apresenta as principais atividades para a engenharia de software com a Coluna 2 que define essas atividades.

| 1. Especificação de software | ( ) O software deve ser validado para garantir que atenda às demandas do cliente. |
| --- | --- |
| 2. Projeto e implementação de software | ( ) O software deve ser produzido para atender às especificações. |
| 3. Validação de software | ( ) O software deve evoluir para atender às necessidades de mudança dos clientes. |
| 4. Evolução de software | ( ) A funcionalidade do software e as restrições a seu funcionamento devem ser definidas. |

Após análise, assinale a alternativa que apresenta a sequência CORRETA:

1. 2, 1, 4, 3.
2. 4, 2, 3, 1.
3. 1, 3, 2, 4.
4. 3, 2, 4, 1.