

# Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023



**Curso:** T-ADS

**Matéria:** Análise e desenvolvimento de sistemas

**Aluno:** Arthur Silva Berdusco De Souza / 2ºB ADS Noturno

**Atividade:** Este trabalho consiste na resolução de exercícios de revisão propostos em anexos.

# Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

## **Descrição e Objetivo**

Este trabalho consiste na resolução de exercícios propostos em anexos.

O trabalho deverá ser feito individualmente.

## **Data de Entrega**

O prazo para a entrega do trabalho 01/06.

O trabalho deverá ser entregue em arquivo pdf, jpg ou png, através do link aberto no blackboard. O trabalho deverá conter:

- resolução dos exercícios.

## ANEXO

### EXERCÍCIOS

1. Julgue a questão.

Algumas aplicações da UML podem ser:

“Ela é uma linguagem padrão para a elaboração da estrutura de projetos de software. Pode ser utilizada para a aplicação, a visualização, a especificação, a construção e a documentação de artefatos que façam uso de sistemas complexos de software”.

a) Certo

b) Errado

2. Considerando o questionamento sobre a importância de se fazer uma distinção entre o desenvolvimento dos requisitos de usuário e os requisitos de sistema no processo de engenharia de requisitos.

Julgue a resposta abaixo.

“Porque o usuário pode exigir características que não sejam compatíveis com o sistema em si, dependendo de seus requisitos. Portanto, é importante fazer a distinção (e também, comparação) entre ambos”.

a) Certo

b) Errado

*Considerando que um teste – ou um processo de teste – consiste em uma sequência de ações executadas com o objetivo de encontrar problemas no software, o que aumenta a percepção de qualidade geral do software e garante que o usuário final tenha um produto que atenda às suas necessidades (PINHEIRO, 2015).*

*Julgue as questões 3 e 4.*

3. “Os testes não podem demonstrar se o software é totalmente livre de defeitos ou se ele se comportará conforme especificado em qualquer situação visto que é sempre possível que um teste que tenha sido esquecido seja aquele que poderia descobrir mais problemas no sistema”.

a) Certo

b) Errado

4. “O Teste é justamente destinado a mostrar que um programa faz o que é proposto a fazer e para descobrir os defeitos do programa antes do uso, os resultados do teste são verificados à procura de

## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, Centro Universitário SENAC 05/2023

erros, anomalias, ou informações sobre atributos não funcionais do programa”.

a) Certo

b) Errado

5. Em relação aos fundamentos da atividade de teste, assinale a alternativa que contém expressões que completam corretamente as lacunas na frase a seguir.

“O processo de teste consiste em executar um programa com o objetivo de revelar a presença de \_\_\_\_\_; ou, falhando nesse objetivo, aumentar a \_\_\_\_\_ sobre o programa. Após revelada a presença do defeito, o processo de \_\_\_\_\_ auxilia em sua busca e correção.”

a) funções recursivas; performance; teste.

b) defeitos; exatidão; teste exaustivo.

c) casos de teste; confiança; apuração.

d) defeitos; confiança; depuração.

e) defeitos; exatidão; casos de teste.

6. Os modelos abaixo compõem os diagramas estruturais?

“Diagramas de Objetos, Diagrama de Estrutura Composta, Diagrama de Classes e Diagrama de Pacotes”.

a) Certo

b) Errado

7. Qual é o modelo ideal que se aplica à maioria dos projetos de software?

Não há um modelo de desenvolvimento de software ideal para a maioria dos projetos, pois a escolha depende de vários fatores. Diferentes modelos, como cascata, incremental, espiral e ágil, são utilizados com base nas necessidades específicas de cada projeto.

8. A afirmação abaixo a respeito do teste de stress está correta?

“Teste de stress é realizado para submeter o software a situações extremas. Basicamente, o teste de stress baseia-se em testar os limites do software e avaliar seu comportamento. Assim, avalia-se até quando o software pode ser exigido e quais as falhas (se existirem) decorrentes do teste”.

a) Certo

b) Errado

9. A afirmação abaixo a respeito do teste de stress está correta?

# Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

“A modelagem é uma visualização dos requisitos, o desejo do cliente do que deve ser implementado, auxilia no entendimento do que o cliente solicitou e do que a equipe deve desenvolver”.

a) Certo

**b) Errado**

10. A afirmação abaixo a respeito do teste de stress está correta?

“Algumas das atividades de gerenciamento de projeto são:

- Identificação de requisitos.
- Abordagem das diferentes necessidades, preocupações e expectativas das partes interessadas no planejamento e execução do projeto.
- Estabelecimento, manutenção e execução de comunicações ativas, eficazes e colaborativas entre os envolvidos no projeto.
- Gerenciamento das partes interessadas visando o atendimento aos requisitos do projeto e a criação das suas entregas.
- Equilíbrio das restrições conflitantes do projeto”.

a) Certo

**b) Errado**

11. Dê um exemplo de diagramas de classe UML, para representar um sistema de software.

Classe: Carro

Atributos:

- marca: string
- modelo: string
- ano: int

Métodos:

- + acelerar(): void
- + frear(): void
- + ligarFarol(): void

12. Descreva por que é importante fazer uma distinção entre o desenvolvimento dos requisitos de usuário e os requisitos de sistema no processo de engenharia de requisitos.

a distinção entre os requisitos de usuário e os requisitos de sistema ajuda a garantir uma compreensão precisa das necessidades dos usuários, a especificação detalhada do sistema, o alinhamento entre expectativas e capacidades, e facilita a rastreabilidade e

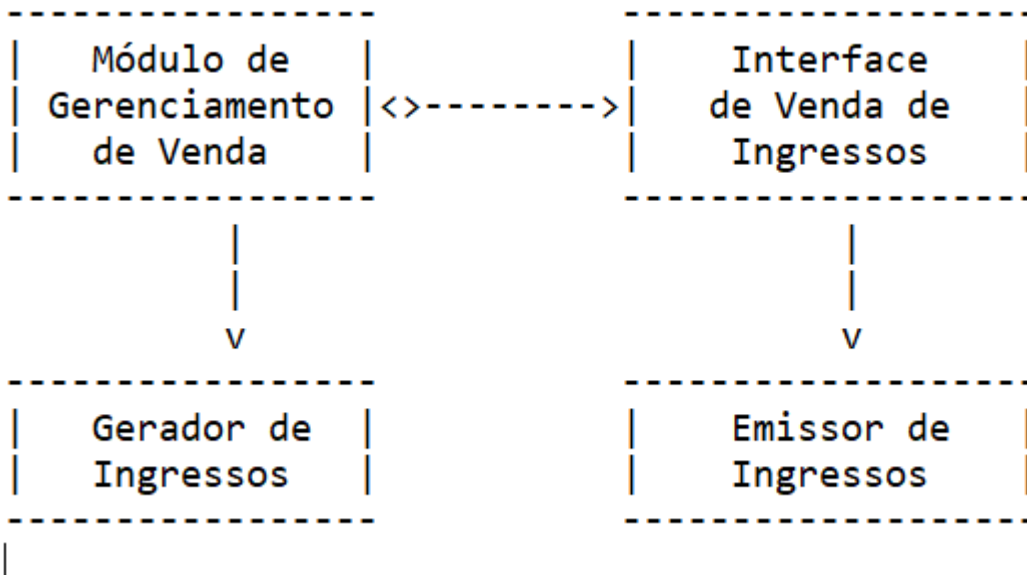
## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

verificação dos requisitos durante o processo de desenvolvimento. Isso contribui para um processo de engenharia de requisitos mais eficiente e um produto final que atenda às expectativas dos usuários.

13. Desenvolva o diagrama de componentes para um sistema de controle de cinema sabendo que:

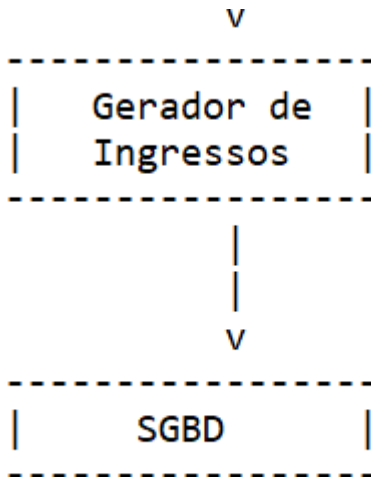
a) É preciso existir um módulo para gerir a venda de ingresso aos clientes. Esse módulo deve gerar os ingressos e emití-los por meio da interface (que pode ser física também) para os clientes do cinema.



## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

b) O sistema necessita de um SGBD para persistir suas informações.



c) Finalmente, existe a necessidade de um módulo de manutenção do sistema, onde basicamente serão mantidos os cadastros de sessões, salas, filmes, atores, gêneros etc.?

O "Módulo de Manutenção do Sistema" é responsável pela gestão dos cadastros, como sessões, salas, filmes, atores e gêneros. Ele se comunica com o SGBD para acessar e atualizar as informações dos cadastros, e a interface de manutenção permite aos usuários adicionar, editar ou excluir registros.

Explique por que os testes podem detectar apenas a presença de erros, e não sua ausência.

14. O que é e para que serve o teste de software?

Os testes de software detectam erros, falhas ou comportamentos indesejados, mas não podem garantir a ausência total de erros no software.

15. Conforme estudamos em sala de aula, por que apenas recentemente houve o foco na arquitetura de software?

Houve um aumento do foco na arquitetura de software devido à complexidade crescente dos sistemas e à necessidade de sistemas mais robustos, escaláveis, seguros e fáceis de manter.

16. O que é arquitetura de aplicações software?

A arquitetura de aplicações de software é a estrutura geral e a organização de um sistema de software, definindo componentes, interações e padrões. Ela facilita o design, a manutenção e a evolução do software, além de auxiliar na comunicação e compreensão da equipe de desenvolvimento.

## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

17. Um paradigma é um estilo de programação, um modelo, uma metodologia. Não se trata de uma linguagem, mas a forma como você soluciona problemas usando uma determinada linguagem de programação.

Sobre paradigma de programação, analise as assertivas e assinale a alternativa que aponta o(s) três tipos de paradigmas de programação estudados em aulas.

I) Paradigma de programação estruturada.

II) Paradigma de programação do tipo funcional

III) Paradigma de programação orientada a objetos.

IV) Paradigma de programação de sistemas.

Os três tipos de paradigmas de programação estudados em aulas são?

a) I e III

b) I, II, III

c) III e IV

d) I, II, III e IV

e) Apenas III

19. A Programação extrema (eXtreme Programming), ou simplesmente XP, é um dos principais métodos ágeis de desenvolvimento de software. Sobre programação extrema (XP), é correto afirmar que:

a) tem a programação em pares como uma de suas práticas.

b) tem como valores: comunicação, simplicidade, feedback, coragem e planejamento.

c) a programação extrema é melhor adaptada para grandes times de desenvolvimento. Esse é um dos princípios básicos da Programação extrema.



## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

*d) o jogo de planejamento da programação extrema é realizado sempre no início de cada projeto e define as fases do RUP e os diagramas da UML que serão utilizados durante o desenvolvimento do projeto.*

*e) uma das principais práticas do XP é a Integração Contínua (Continuous Integration). Esta prática indica que sempre e somente ao final da iteração as novas funcionalidades produzidas devem ser integradas à versão atual do sistema.*

20. Considere as seguintes afirmações sobre princípios ou práticas da XP (Extreme Programming).

*I – Um representante do usuário final do sistema (cliente) deve estar disponível todo o tempo à equipe de XP. Em um processo de Extreme Programming, o cliente é um membro da equipe de desenvolvimento e é responsável por levar ao grupo os requisitos de sistema para implementação.*

*II – Todos os desenvolvedores devem refatorar o código continuamente, assim que encontrarem oportunidades de melhorias de código.*

*III – Os desenvolvedores trabalham em todas as áreas do sistema, de modo que não se desenvolvam ilhas de expertise. Todos os desenvolvedores têm responsabilidade em relação ao código; qualquer um pode mudar qualquer coisa.*

Quais estão corretas?

a) Apenas I.

**b) Apenas I e II.**

c) Apenas I e III.

d) Apenas II e III.

e) I, II e III.

21. Não é recomendável o uso de um método ágil para o desenvolvimento de um sistema de software quando os sistemas forem grandes e também para engenharia de sistemas críticos, devido à necessidade de proteção, segurança e análise de confiança que exigem modificações significativas nos métodos ágeis antes que possam ser usados.

Considerando a afirmação acima, assinale a alternativa correta.

**a) Certo**

## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

b) Errado

22. Os princípios básicos dos métodos ágeis fazem com que os clientes se envolvam no processo de desenvolvimento, o software é desenvolvido em incrementos com o cliente, especificando os requisitos para serem incluídos em cada um, os membros da equipe desenvolvem suas próprias habilidades de trabalho, sem processos prescritivos, tendo em mente que os requisitos irão mudar, com isso projetam o sistema de maneira a aceitar essas mudanças e focalizam a simplicidade, tanto do software a ser desenvolvido quanto do processo do desenvolvimento, procurando ativamente trabalhar para eliminar a complexidade do sistema.

Considerando a afirmação acima, assinale a alternativa correta.

a) Certo

b) Errado

23. Considerando a afirmação abaixo, preencha as lacunas corretamente completando com ágil ou tradicional.

Metodologia \_\_ágil\_\_ é uma maneira para fazer com que as entregas sejam aceleradas durante o desenvolvimento dos projetos. Isso acontece a partir do fracionamento das entregas, utilizando times que sejam auto-organizados e que tenham Inteligência Coletiva.

As Metodologias \_\_ágeis\_\_ são um conjunto de técnicas que contribuem para o processo de gestão de projetos com maior agilidade, interação e entregas com prazos curtos. Através dessas abordagens, os resultados são mensurados por meio de um produto finalizado.

O modelo \_\_ágil\_\_ é baseado em ciclos iterativos e incrementais, trazendo flexibilidade e adaptabilidade aos projetos.

A metodologia \_\_tradicional\_\_ tem etapas bem definidas sendo o planejamento do projeto, uma estimativa em termos de prazo e orçamento, a execução e entrega no final.

A metodologia \_\_tradicional\_\_ se baseia em uma estratégia essencialmente documentada, na qual o planejamento define toda a ordem do desenvolvimento do projeto.

## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

24. A Programação extrema (eXtreme Programming), ou simplesmente XP, é um dos principais métodos ágeis de desenvolvimento de software. Sobre programação extrema (XP), é correto afirmar que:

a) tem a programação em pares como uma de suas práticas.

b) tem como valores: comunicação, simplicidade, feedback, coragem e planejamento.

c) a programação extrema é melhor adaptada para grandes times de desenvolvimento. Esse é um dos princípios básicos da Programação extrema.

d) o jogo de planejamento da programação extrema é realizado sempre no início de cada projeto e define as fases do RUP e os diagramas da UML que serão utilizados durante o desenvolvimento do projeto.

e) uma das principais práticas do XP é a Integração Contínua (Continuous Integration). Esta prática indica que sempre e somente ao final da iteração as novas funcionalidades produzidas devem ser integradas à versão atual do sistema. Mas isso não quer dizer que não é documentado, não planejado, assim como na tradicional.

25. Em programação orientada a objetos, um objeto é um conceito que encapsula, em sua estrutura, dados e código. A esses elementos damos o nome, respectivamente, de:

a) Atributos e métodos

b) Informações e execuções

c) Teoria e prática

d) Ações e características

26. À especificação que define os atributos e métodos que estarão presentes em um determinado tipo de objeto damos o nome de:

a) Instância

b) Projeto

c) Classe

d) Coletivo

27. Em programação orientada a objetos, uma mensagem é:

## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

a) A forma de comunicação entre a classe e a sua instância resultando na contagem do número de objetos que compartilham o mesmo tipo.

b) A forma de comunicação entre os diversos objetos presentes no sistema e resulta na chamada de um método pelo objeto receptor da mensagem

c) A forma de comunicação entre um objeto do sistema com objetos de outros sistemas através de chamadas à sua interface.

d) A forma de comunicação entre o sistema e o usuário utilizada para coletar dados via entrada padrão

28. Durante o processo de criação de um sistema orientado a objetos, precisamos representar computacionalmente as entidades do domínio do problema. Essa representação é realizada removendo detalhes menos importantes e focando nos aspectos principais de cada entidade e a esse processo damos o nome de:

a) Abstração

b) Instanciação

c) Resumo

d) Materialização

29. Assinale a alternativa que possui a afirmação incorreta sobre o uso de herança em programação orientada a objetos:

a) A utilização de herança permite o reaproveitamento de código, diminuindo a redundância na implementação do sistema

b) A utilização de herança ajuda a organizar as abstrações presentes no código de maneira hierárquica

c) Herança é uma relação entre duas classes que possuem um relacionamento do tipo "é-um(a)" entre si

d) Utilizamos o recurso de herança em orientação a objetos quando queremos passar valores de variáveis que estão em um objeto que será destruído para outro objeto

30. Selecione a alternativa que preenche corretamente as lacunas na afirmação abaixo

"Denominamos \_\_\_\_\_ a classe que herda os atributos e métodos em uma relação de herança e \_\_\_\_\_ a classe da qual se herda."

## Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

**a) subclasse - superclasse**

- b) herdeira - inventariado
- c) superclasse - hiperclasse
- d) superclasse - subclasse

31. Uma das vantagens da programação orientada a objetos é o uso da herança, que possibilita a utilização de uma classe para criar outras a partir dela. A herança só pode ser aplicável quando se pode dizer “é um”, ou seja, quando um objeto é do mesmo tipo que outro.

Ao implementarmos uma classe B que é subclasse de A:

a) **Podemos implementar novos métodos e modificar os métodos herdados de A**

- b) Não podemos implementar novos métodos nem modificar os métodos herdados de A
- c) Podemos modificar os métodos herdados de A, mas não podemos implementar novos métodos
- d) Podemos implementar novos métodos, mas não podemos modificar os métodos herdados de A

# Exercícios: ADS-Análise e Desenvolvimento de Sistema

Prof. Marcello, **Centro Universitário SENAC** 05/2023

Do ponto de vista de Software e Engenharia de Software responda as questões abaixo.

## 1) O que é Software e Engenharia de Software?

Software de computador é o produto que profissionais de software desenvolvem e ao qual dão suporte no longo prazo. Abrange programas executáveis em um computador de qualquer porte ou arquitetura, conteúdos (apresentados à medida que os programas são executados), informações descritivas tanto na forma impressa (*hard copy*) como na virtual, abrangendo praticamente qualquer mídia eletrônica. A engenharia de software abrange um processo, um conjunto de métodos (práticas) e um leque de ferramentas que possibilitam aos profissionais desenvolverem software de altíssima qualidade.

## 2) Quem desenvolve o Software?

Os engenheiros de software criam e dão suporte a ele e, praticamente, todos do mundo industrializado o utilizam, direta ou indiretamente.

## 3) Por que o Software é importante?

Software é importante porque afeta a quase todos os aspectos de nossas vidas e tornou-se pervasivo (incorporado) no comércio, na cultura e em nossas atividades cotidianas. A engenharia de software é importante porque ela nos capacita para o desenvolvimento de sistemas complexos dentro do prazo e com alta qualidade.

## 4) Quais são as etapas envolvidas?

Cria-se software para computadores da mesma forma que qualquer produto bem-sucedido: aplicando-se um processo adaptável e ágil que conduza a um resultado de alta qualidade, atendendo às necessidades daqueles que usarão o produto. Aplica-se uma abordagem de engenharia de software.

## 5) Qual é o artefato utilizado? (Do ponto de vista de um engenheiro de software e ponto de vista do usuário)

Do ponto de vista de um engenheiro de software, é um conjunto de programas, conteúdo (dados) e outros artefatos que são software. Porém, do ponto de vista do usuário, o artefato consiste em informações resultantes que, de alguma forma, tomam a vida dele melhor.

## 6) Como garantir que o trabalho foi feito corretamente?

Selecione as ideias que são aplicáveis ao software que você desenvolver e use-as em sua resposta.