



ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

MATERIAL INSTRUCIONAL ESPECÍFICO

TOMO 15

CQA/UNIP – Comissão de Qualificação e Avaliação da UNIP

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

MATERIAL INSTRUCIONAL ESPECÍFICO

TOMO 15

Christiane Mazur Doi

Doutora em Engenharia Metalúrgica e de Materiais, Mestra em Ciências - Tecnologia Nuclear, Especialista em Cálculo e Matemática Aplicada, Especialista em Língua Portuguesa e Literatura, Especialista em Recursos Gramaticais para Revisão Textual, Engenheira Química e Licenciada em Matemática, com Aperfeiçoamento em Estatística e MBA em Engenharia Econômica. Professora titular da Universidade Paulista.

Larissa Rodrigues Damiani

Doutora e Mestra em Ciências pelo programa de Engenharia Elétrica – Microeletrônica e Engenharia Elétrica - modalidade Eletrônica (ênfase em Telemática). Professora titular da Universidade Paulista.

Material instrucional específico, cujo conteúdo integral ou parcial não pode ser reproduzido nem utilizado sem autorização expressa, por escrito, da CQA/UNIP – Comissão de Qualificação e Avaliação da UNIP – UNIVERSIDADE PAULISTA.

Questão 1

Questão 1.¹

Empreender em computação não se resume apenas ao fato de ter uma ideia sobre um aplicativo, implementá-la e disponibilizá-la aos seus usuários finais. Esse conceito vai além, é preciso analisar uma série de variáveis, como questões sociais e humanísticas, profissionais, ambientais, legais, políticas e tecnológicas. Entre as ações necessárias para empreender em computação, é imprescindível observar a legislação vigente. No Brasil, a Lei Geral de Proteção de Dados Pessoais (LGPD), Lei Nº 13.709/2018, dispõe sobre o tratamento de dados pessoais, inclusive nos meios digitais, por pessoa natural ou por pessoa jurídica de direito público ou privado, com o objetivo de proteger os direitos fundamentais de liberdade e de privacidade e o livre desenvolvimento da personalidade da pessoa natural.

Acerca das providências dispostas no art. 2º da LGPD, assinale os fundamentos da disciplina da proteção de dados pessoais.

- A. O direito ao usuário; das obrigações das organizações nacionais ou estrangeiras; obrigatoriedade do consentimento; a livre iniciativa, a livre concorrência e a defesa do consumidor; os direitos humanos, o livre desenvolvimento da personalidade, a dignidade e o exercício da cidadania pelas pessoas naturais e regulação da Autoridade Nacional de Proteção de Dados Pessoais (ANPD).
- B. O direito à revogação de um consentimento; a transferência de dados para outro fornecedor de serviços; a solicitação de remoção de dados pessoais; a inviolabilidade da intimidade, da honra e da imagem; o desenvolvimento econômico e tecnológico e inovação; a preservação da vida e da integridade física de uma pessoa.
- C. O respeito à privacidade; obrigatoriedade do consentimento; os deveres e as obrigações dos usuários finais; os deveres e as obrigações das organizações nacionais ou estrangeiras; a observação ao código de defesa do consumidor; e a obrigatoriedade do respeito à honra e à imagem.
- D. O direito ao anonimato; a liberdade de expressão; a inviolabilidade da intimidade; a adoção de medidas preventivas de segurança; o direito ao recurso e à revisão; a administração de riscos e falhas; o estabelecimento de agentes de tratamento; e a regulação da Autoridade Nacional de Proteção de Dados Pessoais (ANPD).
- E. O respeito à privacidade; a autodeterminação informativa; a liberdade de expressão, de informação, de comunicação e de opinião; a inviolabilidade da intimidade, da honra e da

¹Questão 26 – Enade 2021.

imagem; o desenvolvimento econômico e tecnológico e a inovação; a livre iniciativa, a livre concorrência e a defesa do consumidor; e os direitos humanos, o livre desenvolvimento da personalidade, dignidade e o exercício da cidadania pelas pessoas naturais.

1. Análise da questão

A questão exige conhecimentos acerca do conteúdo disposto no artigo 2º da Lei Geral de Proteção de Dados Pessoais (LGPD), Lei Nº 13.709/2018, relevante para aqueles que desejam empreender na área de tecnologia da informação. Vamos estudar um pouco a respeito do seu conteúdo a seguir.

Lei Geral de Proteção de Dados Pessoais (LGPD)

De forma geral, a LGPD dispõe sobre o tratamento de dados pessoais em quaisquer meios, inclusive nos meios digitais, tanto por pessoa natural quanto por pessoa jurídica. Seus objetivos são proteger os direitos fundamentais de liberdade e de privacidade e o livre desenvolvimento da personalidade da pessoa natural. Nesse contexto, o termo “pessoa natural” refere-se a qualquer ser humano vivo que pode titularizar direitos e assumir obrigações. Já o termo “dado pessoal” refere-se a qualquer informação relacionada à pessoa natural.

Em seu artigo 2º, a LGPD elenca os fundamentos da disciplina da proteção de dados pessoais. Esses fundamentos são:

- I - o respeito à privacidade;*
- II - a autodeterminação informativa;*
- III - a liberdade de expressão, de informação, de comunicação e de opinião;*
- IV - a inviolabilidade da intimidade, da honra e da imagem;*
- V - o desenvolvimento econômico e tecnológico e a inovação;*
- VI - a livre iniciativa, a livre concorrência e a defesa do consumidor; e*
- VII - os direitos humanos, o livre desenvolvimento da personalidade, a dignidade e o exercício da cidadania pelas pessoas naturais.*

Um ponto importante do texto da LGPD é o consentimento. O consentimento do cidadão é a base para que dados pessoais possam ser tratados, salvo algumas exceções justificadas, como tratar dados indispensáveis ao cumprimento de uma obrigação legal.

A LGPD traz várias garantias ao cidadão, que pode solicitar que dados sejam deletados, revogar um consentimento e transferir dados para outro fornecedor de serviços, entre outras

ações. Além disso, o tratamento dos dados deve ser feito levando em conta alguns itens, como finalidade e necessidade, que devem ser previamente informados à pessoa natural.

2. Análise das alternativas

A – Alternativa incorreta.

JUSTIFICATIVA. Os seguintes itens não constam como fundamentos da disciplina da proteção de dados pessoais, de acordo com o artigo 2º da LGPD: o direito ao usuário; das obrigações das organizações nacionais ou estrangeiras; obrigatoriedade do consentimento; e regulação da Autoridade Nacional de Proteção de Dados Pessoais (ANPD). Esses temas são abordados em outros artigos da lei.

B – Alternativa incorreta.

JUSTIFICATIVA. Os seguintes itens não constam como fundamentos da disciplina da proteção de dados pessoais, de acordo com o artigo 2º da LGPD: o direito à revogação de um consentimento; a transferência de dados para outro fornecedor de serviços; a solicitação de remoção de dados pessoais; e a preservação da vida e da integridade física de uma pessoa. Esses temas são abordados em outros artigos da lei.

C – Alternativa incorreta.

JUSTIFICATIVA. Os seguintes itens não constam como fundamentos da disciplina da proteção de dados pessoais, de acordo com o artigo 2º da LGPD: obrigatoriedade do consentimento; os deveres e as obrigações dos usuários finais; os deveres e as obrigações das organizações nacionais ou estrangeiras; e a observação ao código de defesa do consumidor. Esses temas são abordados em outros artigos da lei.

D – Alternativa incorreta.

JUSTIFICATIVA. Os seguintes itens não constam como fundamentos da disciplina da proteção de dados pessoais, de acordo com o artigo 2º da LGPD: o direito ao anonimato; a adoção de medidas preventivas de segurança; o direito ao recurso e à revisão; a administração de riscos e falhas; o estabelecimento de agentes de tratamento; e regulação da Autoridade Nacional de Proteção de Dados Pessoais (ANPD). Esses temas são abordados em outros artigos da lei.

E – Alternativa correta.

JUSTIFICATIVA. O texto da alternativa corresponde aos fundamentos da disciplina da proteção de dados pessoais, de acordo com o artigo 2º da LGPD.

3. Indicações bibliográficas

- BRASIL. Lei Nº 13.709, de 14 de agosto de 2018. *Lei Geral de Proteção de Dados Pessoais (LGPD)*. Disponível em <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/L13709compilado.htm>. Acesso em 19 fev. 2025.
- BRASIL. *O que muda com a LGPD*. Disponível em <<https://www.serpro.gov.br/lgpd/menu/a-lgpd/o-que-muda-com-a-lgpd>>. Acesso em 19 fev. 2025.

Questão 2

Questão 2.²

Uma secretaria municipal de esportes decidiu criar e disponibilizar um site que sirva como portal de comunicação com pessoas da terceira idade. A finalidade é fornecer orientações que propiciem práticas para uma vida mais saudável, além de divulgar eventos esportivos ofertados pelo município para esse público-alvo. O projeto ainda prevê a interação entre os usuários do site, por meio da publicação de fotos, que podem ser curtidas e comentadas.

A iniciativa de criação do site foi embasada em uma pesquisa mostrando que a maioria da população de terceira idade do município dispõe de acesso à internet, especialmente por meio de dispositivos móveis (smartphones e tablets), embora uma parcela dessa população prefira realizar o acesso com computadores de mesa e notebooks.

Considerando as informações do texto e as responsabilidades inerentes ao desenvolvimento do site, avalie as afirmativas.

- I. A segurança no uso é um fator crítico de usabilidade a ser considerado no projeto de interface para o site, tendo em vista que pessoas da terceira idade são temerosas quanto ao registro de dados pessoais e confidenciais em sites e aplicativos.
- II. A responsividade, que pressupõe o uso de valores relativos para definir as dimensões de elementos na estilização de páginas HTML, deve ser considerada na construção do site, tendo em vista que o acesso deve ocorrer por meio de dispositivos com diferentes tamanhos de tela.
- III. A construção de protótipos pode antecipar problemas de design de interação e minimizar custos de desenvolvimento e, por isso, é viável para estabelecer uma comunicação mais objetiva com o público-alvo e assim possibilitar a validação de usabilidade e da experiência do usuário.
- IV. O design participativo é um processo de design de Interação Humano-Computador (IHC) viável para direcionar o projeto de interfaces e de experiência com o usuário do site, pois o público-alvo tende a apresentar dificuldades no manuseio de tecnologias da informação e da comunicação.

É correto apenas o que se afirma em

- | | | |
|-----------------|------------------|-------------|
| A. I e II. | B. I e III. | C. II e IV. |
| D. I, III e IV. | E. II, III e IV. | |

²Questão 20 – Enade 2021.

1. Análise da questão

A questão testa conhecimentos a respeito da interface humano-computador, aplicada a websites. Entre os temas abordados, estão o design de interação, a responsividade, a segurança de dados e o processo de design participativo. Esses temas serão tratados a seguir.

1.1. Usabilidade e experiência do usuário

A Interface Humano-Computador (IHC) é uma área de estudo que se concentra no design, na criação e na implementação de interfaces interativas para sistemas de software. Ela estuda a forma como os humanos interagem com os computadores, de maneira a projetar tecnologias que sejam intuitivas, fáceis e agradáveis. Além de profissionais da área de software, o estudo da IHC também engloba profissionais de diversas outras áreas, como psicologia, sociologia e ergonomia.

Os princípios da IHC são aplicados em muitos tipos de software, como aplicativos de desktop, websites, aplicativos móveis e sistemas de realidade virtual. A IHC também considera fatores como acessibilidade e inclusão, garantindo que as tecnologias sejam utilizáveis por pessoas com diferentes necessidades.

Nesse contexto, o **design de interação** é a área que se preocupa em estudar as melhores formas de garantir ao usuário usabilidade e, ao mesmo tempo, uma boa experiência na interação.

A **usabilidade** diz respeito à capacidade de um produto de software ser usável. Em linhas gerais, ser usável significar ser fácil, eficiente e útil para o usuário. A **experiência do usuário**, por sua vez, diz respeito às sensações e às emoções que os usuários vivenciam ao utilizar um produto de tecnologia. Um bom design de interação é aquele que alia usabilidade e boa experiência.

Um website, por exemplo, pode ser bom em termos de usabilidade, mas frustrante em termos de experiência do usuário. Se um website de consulta a livros digitais permite que achemos o que precisamos com facilidade, podemos dizer que, em termos de usabilidade, ele tem um bom design. Contudo, sua interface pode ser apática e não trazer uma experiência agradável.

1.2. Responsividade

A **responsividade** é um conceito no design web que se refere à capacidade de um site se adaptar automaticamente ao tamanho da tela do dispositivo que o usuário está usando para visualizá-lo. Isso significa que será fácil usar o site e navegar nele em diversos dispositivos, como desktops, notebooks, smartphones ou tablets.

A responsividade é alcançada pelo uso de técnicas de design fluido, como o emprego de grades fluidas e de imagens flexíveis. As grades fluidas definem larguras usando taxas percentuais em vez de pixels fixos, permitindo que os elementos da página se redimensionem proporcionalmente ao dispositivo que os exibe. As imagens flexíveis também são redimensionadas automaticamente para caber nas telas, além de serem reorientadas de acordo com a posição do dispositivo. Além disso, podem ser aplicados tratamentos de textos e rearranjo de menus.

Um website responsivo, portanto, deve garantir o seu acesso no maior número de dispositivos possível, adaptando-se a eles e proporcionando melhor experiência ao usuário.

1.3. Segurança de dados

No contexto da tecnologia da informação, a **segurança de dados** refere-se à proteção de informações confidenciais de um sistema de software e à prevenção de acesso não autorizado a essas informações. Isso é especialmente importante quando os usuários fornecem seus dados pessoais. A segurança de dados sempre teve grande importância, mas ganhou destaque após a chegada da Lei Geral de Proteção de Dados Pessoais (LGPD).

Se os usuários não confiam na segurança de um website, eles podem hesitar em usar o site ou em fornecer suas informações. Por isso, a segurança de dados, além de garantir um ambiente seguro, pode ser benéfica para a experiência dos usuários.

As chances de fidelizar os usuários são maiores se, com a segurança de dados, a empresa investir na usabilidade e na experiência dos websites, pois pouco adianta investir na segurança de dados se o uso do sistema não for amigável, fluido nem agradável.

1.4. Design participativo

Entre os diversos tipos de processo de design de IHC, os principais são aqueles que atraem os usuários para trabalhar com os designers. Um processo de design que conta com a participação do usuário desde o projeto tende a obter mais sucesso na sua fase de implantação, pois o designer tem condições de ser mais assertivo, em razão do seu alto grau de entendimento das necessidades de quem vai utilizar a interface.

Nesse cenário, profissionais de outras áreas de interesse ao projeto ou usuários finais do produto de software podem colaborar com o projeto de design como parte da equipe de trabalho. Essa abordagem, portanto, vai além de coletar informações para os designers e envolve a participação efetiva das partes interessadas (stakeholders).

A prototipagem é uma ferramenta valiosa no design participativo. A construção de protótipos pode antecipar problemas de design de interação, possibilitando que as ideias para o design sejam pensadas e elaboradas antes que seja preciso gastar tempo e recursos financeiros desenvolvendo-as diretamente na interface final. Esse processo permite que as ideias sejam aperfeiçoadas e que soluções para eventuais problemas sejam descobertas.

O protótipo deve ser avaliado pelos usuários. Isso permite a revisão e a melhora do projeto de forma colaborativa e iterativa, até que seja atingida uma solução satisfatória.

2. Análise das afirmativas

I – Afirmativa incorreta.

JUSTIFICATIVA. A segurança não é considerada um fator crítico de usabilidade. A usabilidade refere-se à facilidade de uso de um site, incluindo um design intuitivo da interface. Por sua vez, a segurança refere-se à proteção de informações confidenciais e à prevenção de acesso não autorizado.

II – Afirmativa correta.

JUSTIFICATIVA. O website descrito no enunciado será acessado por diferentes dispositivos, como smartphones, tablets, computadores de mesa e notebooks. Por isso, a responsividade é uma característica importante a ser considerada no projeto, já que ela garantirá que o site poderá se adaptar a diversos dispositivos.

III – Afirmativa correta.

JUSTIFICATIVA. A construção de protótipos pode antecipar problemas de design de interação e minimizar custos de desenvolvimento. O protótipo deve ser apresentado aos usuários, o que garante comunicação direta com o público-alvo do website.

IV – Afirmativa correta.

JUSTIFICATIVA. O público-alvo do site é formado por pessoas da terceira idade. Esse público, de forma geral, tende a apresentar dificuldades no manuseio de tecnologias da informação e da comunicação. Nesse caso, o design participativo é um processo de design viável, pois permitirá aos designers compreenderem, de forma assertiva, as dificuldades e as expectativas dos usuários. Isso fará com que a usabilidade e a experiência sejam aprimoradas.

Alternativa correta: E.

3. Indicações bibliográficas

- BARRETO, J. S. et al. *Interface Humano-Computador*. Porto Alegre: SAGAH, 2018.
- INSTITUTO PROPAGUE. *Segurança de dados: o que é e como ela impacta a experiência do usuário*. Disponível em <<https://institutopropague.org/tecnologia-e-dados/seguranca-de-dados-o-que-e-e-como-ela-impacta-a-experiencia-do-usuario/>>. Acesso em 19 fev. 2024.
- PAZ, M. *Webdesign*. Curitiba: InterSaberes, 2021.

Questão 3**Questão 3.**³

Suponha que sua faculdade tendo, em sua rede de computadores, máquinas que compartilham arquivos e serviços, contratou você para uma consultoria de TI. Atualmente, na sala dos professores, existem 3 máquinas ligadas à rede, com as configurações conforme a imagem a seguir. Porém, a estação de trabalho “PROFESSORES01” não consegue enviar documentos para impressão em uma impressora, corretamente instalada e compartilhada na estação de trabalho “PROFESSORES03”.

 PROFESSORES01	ENDEREÇO IP: 175.16.10.200 MÁSCARA DE SUB-REDE: 255.255.255.0 GATEWAY: 175.16.1.1
 PROFESSORES02	ENDEREÇO IP: 175.16.20.200 MÁSCARA DE SUB-REDE: 255.255.0.0 GATEWAY: 175.16.1.1
 PROFESSORES03	ENDEREÇO IP: 175.16.30.200 MÁSCARA DE SUB-REDE: 255.255.0.0 GATEWAY: 175.16.1.1

Nesse caso, que alteração deve ser realizada para que o problema seja resolvido?

- A. Alterar o gateway da estação PROFESSORES03 para 175.16.10.200.
- B. Alterar o endereço de IP da estação PROFESSORES01 para 175.16.30.200.
- C. Alterar a máscara de sub-rede da estação PROFESSORES01 para 255.255.0.0.
- D. Alterar a máscara de sub-rede da estação PROFESSORES03 para 255.255.255.0.
- E. Alterar o gateway da estação PROFESSORES01 para o endereço de IP 175.16.30.200.

³Questão 30 – Enade 2021.

1. Análise da questão

A questão testa conhecimentos básicos a respeito de redes de computadores, em especial, a respeito do endereço de protocolo de internet (IP), da máscara de sub-rede e do gateway de estações de trabalho interligadas na mesma rede local. Abordaremos esses tópicos a seguir, de forma breve.

Endereço IP, máscara de sub-rede e gateway

No contexto de redes locais de computadores (LAN), o **endereço IP** é um código numérico exclusivo, formado por 32 bits, que identifica um dispositivo em uma LAN. Esse código é formado por quatro octetos (ou seja, quatro conjuntos de 8 bits), separados entre si por um ponto. O código 192.158.1.38 é um exemplo de endereço IP.

Cada octeto é um número expresso em sistema decimal, que pode variar de 0 a 255. O valor decimal 0 corresponde ao padrão de bits 00000000, enquanto o valor decimal 255 corresponde ao padrão de bits 11111111.

O endereço IP é separado entre porção de rede e porção da estação. Para isso, a **máscara de sub-rede** é um código numérico usado para determinar onde termina o endereço da rede e onde começa o endereço da estação. Devido à forma como a máscara de rede é usada, os bits de um mesmo octeto devem ser definidos com valor "1" da esquerda para direita. Por exemplo, uma máscara de sub-rede 255.0.0.0 corresponde a dizer que o primeiro octeto do endereço IP está sendo utilizado para indicar a rede, e os três octetos restantes estão sendo utilizados para indicar estações pertencentes a essa rede. Já a máscara de sub-rede 255.255.0.0 corresponde a dizer que os dois primeiros octetos estão sendo utilizados para indicar a rede, e os dois últimos, para indicar estações.

O **gateway** é um recurso que atua como intermediário para a troca de informações com dispositivos fora da rede local. No contexto de uma LAN, o valor do gateway geralmente indica o endereço IP do roteador na rede. Quando um dispositivo precisa se comunicar com outro que está fora da sua rede local, ele envia a informação para o gateway (ou seja, para o roteador). O roteador, então, encaminhará essa informação para a rede externa, como a internet.

2. Resolução da questão

Pelas informações dispostas no quadro do enunciado, vemos que os endereços IP de todas as estações têm valores iguais para os dois primeiros octetos (175.16.x.x). Isso nos leva a imaginar que os dois primeiros octetos identificam a rede, que deve ser a mesma para as três estações.

O valor do gateway é o endereço para a comunicação com dispositivos fora da rede local. Todos os dispositivos na mesma LAN devem ter o mesmo valor de gateway para garantir que possam se comunicar corretamente com dispositivos fora da rede. No caso das estações de trabalho da questão, todas elas têm o mesmo valor de gateway (175.16.1.1), o que é correto.

O problema, portanto, reside na configuração da máscara de sub-rede. Perceba que a máscara da estação de trabalho "PROFESSORES01" é 255.255.255.0. Já para as outras duas estações, o código é expresso como 255.255.0.0.

A máscara de sub-rede determina quais endereços IP estão na mesma rede que o dispositivo. Para a estação de trabalho "PROFESSORES01", o código 255.255.255.0 indica que apenas dispositivos com o mesmo endereço IP nos primeiros três octetos (175.16.10.x) estarão na mesma rede dessa estação.

No entanto, a estação de trabalho "PROFESSORES03", que compartilha a impressora, tem endereço IP dado por 175.16.30.200. Isso significa que essa estação está em uma rede diferente de "PROFESSORES01" de acordo com a configuração atual.

Para resolver o problema, a solução mais adequada é alterar a máscara de sub-rede da estação de trabalho "PROFESSORES01" para 255.255.0.0, que é a mesma das outras estações de trabalho. Isso colocará as três estações na mesma rede, permitindo que tanto "PROFESSORES01" quanto "PROFESSORES02" acessem a impressora compartilhada em "PROFESSORES03".

Note que alterar o endereço IP da estação "PROFESSORES01" para 175.16.30.200, conforme sugere a alternativa B, não resolveria o problema, pois a estação "PROFESSORES03" já tem esse endereço. Cada dispositivo em uma rede deve ter um endereço IP único, para evitar conflitos.

Alternativa correta: C.

3. Indicações bibliográficas

- FOROUZAN, B. A.; MOSHARRAF, F. *Redes de computadores: uma abordagem top-down*. Porto Alegre: AMGH, 2013.
- MORAES, A. F. de. *Redes de computadores: fundamentos*. São Paulo: Érica, 2020.

Questão 4

Questão 4.⁴

Leia o texto a seguir.

Um modelo de negócios descreve a lógica de criação, entrega e captura de valor por parte de uma organização. A ferramenta de geração de modelo de negócios Canvas, desenvolvida por Osterwalder, tem sido uma das mais utilizadas, principalmente entre as startups. O Canvas é dividido em nove componentes básicos, que cobrem as quatro áreas principais de um negócio: clientes, oferta, infraestrutura e viabilidade financeira. Os componentes são os seguintes: 1) segmentos de clientes; 2) proposta de valor; 3) canais; 4) relacionamento com cliente; 5) fontes de receita; 6) recursos principais; 7) atividades-chave; 8) parcerias principais; e 9) estrutura de custo.

OSTERWALDER, A.; PIGNEUR, Y. *Business Model Generation - Inovação em Modelos de Negócios: um manual para visionários, inovadores e revolucionários*. Rio de Janeiro: Alta Books, 2011 (com adaptações).

Neste contexto, conclui-se que as startups

- A. financeiras desenvolvem primeiro o plano de negócios, para em seguida partir para o Canvas.
- B. de big data utilizam o Canvas para a modelagem da validação da ideia de negócio, ou seja, do problema a ser resolvido.
- C. de educação dispensam a utilização do Canvas, ou seja, os modelos de negócios deste segmento já nascem validados.
- D. de logística utilizam o plano de negócios em vez do Canvas, ou seja, elas nascem a partir de um produto mínimo viável (MVP).
- E. de impacto social podem desconsiderar o componente 1 (segmentos de clientes), pois possuem somente um segmento.

1. Análise da questão

Para responder à questão, devemos ter conhecimentos básicos a respeito das etapas do processo empreendedor, em especial, das etapas de identificação de oportunidades e de criação do plano de negócios. Além disso, é necessário, também, sabermos o básico a respeito da ferramenta de geração de modelo de negócios Canvas, que é uma ferramenta envolvida no desenvolvimento dessas etapas. Esses temas serão abordados a seguir.

⁴Questão 32 – Enade 2021.

1.1. Processo empreendedor

No campo do empreendedorismo, **oportunidades** podem ser definidas como ideias com potencial de retorno econômico, ou seja, ideias que podem ser transformadas em produtos ou serviços.

Há vários modelos que auxiliam o empreendedor na identificação de oportunidades. Uma das mais conhecidas e utilizadas nos dias atuais é a ferramenta de geração de modelo de negócios Canvas. Com ela, é possível que o empreendedor selecione a oportunidade mais interessante para transformar efetivamente em um negócio. Basicamente, várias ideias são analisadas pelo empreendedor antes de definir quais levarão a oportunidades com maior potencial de execução ou de retorno econômico. Após a análise da oportunidade, por meio de um método como a ferramenta Canvas, o empreendedor pode rever a ideia inicial. Com isso, o profissional encerra a primeira fase do processo empreendedor. As fases desse processo podem ser vistas na figura 1.



Figura 1. Abordagem cíclica do processo empreendedor.

DORNELES, J. *Plano de negócios com o modelo Canvas: guia prático de avaliação de ideias de negócio*. São Paulo: Empreende, 2020, p. 3 (com adaptações).

Após selecionar uma oportunidade, o empreendedor inicia o desenvolvimento do plano de negócios, que é uma etapa crucial antes de se criar uma empresa. Dependendo da abordagem adotada, o empreendedor pode optar por desenvolver um planejamento estruturado, com previsões e projeção de mercado, receita, custos, despesas, o que corresponde ao plano de negócios tradicional. Caso não haja possibilidade de fazer previsões detalhadas, pode-se optar por outro modelo de plano de negócios, mais simplificado e atualizado. Ao longo do processo de desenvolvimento do plano de negócios, o profissional pode sentir a necessidade de rever a ideia ou a oportunidade.

Com a conclusão do plano, o profissional saberá com clareza quais recursos (funcionário, dinheiro, infraestrutura etc.) serão necessários. A etapa 4 do processo empreendedor corresponde, justamente, à etapa em que o profissional deve arrumar e alocar os recursos. Com eles em mãos, já é possível partir para a gestão da empresa.

Após a empresa entrar em operação, o empreendedor possivelmente terá novas ideias, identificará novas oportunidades e, desse modo, precisará desenvolver novos planos de negócios ou ajustar o plano de negócios atual. Assim, o ciclo é iniciado novamente.

1.2. Ferramenta Canvas

A ferramenta de geração de modelo de negócios Canvas tem como objetivo analisar uma ideia e, a partir dela, conceituar um modelo de negócios. Um **modelo de negócios** é a explicação de como a empresa funciona e gera valor a seus clientes. Em essência, ele busca entender como a empresa fará dinheiro, qual será seu modelo de receita e como várias áreas e processos de negócio se relacionam para fazer a empresa funcionar. O modelo de negócios é comumente resumido em um quadro Canvas.

O quadro 1 apresenta o quadro Canvas, com seus nove blocos integrados: segmentos de clientes, proposta de valor, canais, relacionamento com cliente, fontes de receita, recursos principais, atividades-chave, parcerias principais e estrutura de custo.

A ideia é que o empreendedor responda às perguntas de cada bloco, iniciando pelas suas **propostas de valor** e pelos **segmentos de clientes** e, depois, responda às questões dos blocos **canais** e **relacionamentos com o cliente**. Posteriormente, o quadro todo deve ser gradualmente preenchido.

Uma forma de aplicar o Canvas é pela utilização de blocos “post-it”, de maneira a posicionar as respostas no quadro. Assim, o modelo de negócio vai sendo construído visualmente, de maneira cíclica.

Quadro 1. Modelo de negócios Canvas.

Quadro 21: Modelo de negócios Canvas				
Parceiros-chave Quem são seus parceiros-chave? Quem são seus fornecedores-chave? Quais recursos-chave adquirimos de nossos parceiros? Quais atividades nossos parceiros realizam?	Atividades-chave Quais atividades nossa proposta de valor requer? Quais são nossos canais de distribuição? Como é o relacionamento com o cliente? Quais são as fontes de receita?	Propostas de valor Que valores entregamos aos nossos clientes? Quais problemas dos nossos clientes ajudamos a resolver? Que categorias de produtos e serviços oferecemos a cada segmento de clientes? Quais necessidades dos clientes nós satisfazemos? O que/qual é o nosso mínimo produto viável?	Relacionamentos com os clientes Como nós conquistamos, mantemos e aumentamos nossos clientes? Quais relacionamentos com o cliente nós definimos/temos? Como esses relacionamentos estão integrados no nosso modelo de negócio? Qual é o custo envolvido?	Segmentos de clientes Para quem nós criamos valor? Quem são nossos mais importantes clientes? Quais são nossos clientes típicos?
	Recursos-chave Quais recursos-chave nossa proposta de valor requer? Canais, relacionamentos, modelo de receita?		Canais Por meio de quais canais nossos segmentos de clientes querem ser alcançados? Como outras empresas chegam até eles hoje? Quais canais funcionam melhor? Quais canais são mais eficientes em custo? Como promovemos a integração dos canais com a rotina dos clientes?	
Estrutura de custos Quais são os custos mais importantes de nosso modelo de negócio? Quais recursos-chave são os mais caros? Quais atividades-chave são as mais caras?			Fontes de receita Para qual proposta de valor nossos clientes estão dispostos a pagar? O que eles estão comprando hoje? Qual é nosso modelo de receita? Quais são nossas políticas de preços?	

DORNELES, J. *Plano de negócios com o modelo Canvas: guia prático de avaliação de ideias de negócio*. São Paulo: Empreende, 2020, p. 17 (com adaptações).

Ao concluir o quadro Canvas, o profissional terá uma análise de oportunidade em mãos e poderá, então, fazer a pesquisa primária com seu público-alvo. Em seguida, estará preparado para desenvolver o plano de negócios da empresa.

O **plano de negócios** deve mostrar os custos e as despesas, o investimento inicial, a necessidade de recursos para colocar a empresa em operação, a estratégia de crescimento, a estratégia de marketing e de vendas, assim como a projeção de receita e lucro para os períodos seguintes. O desenvolvimento do plano de negócios tradicional pode durar de semanas a meses.

O plano de negócios tradicional justifica-se nos casos em que o profissional tem um objetivo claro a atingir. Com o intuito de focar em algo mais prático e rápido, novos conceitos, como o modelo de negócio Canvas, têm se popularizado, principalmente no mercado de tecnologia da informação. A integração do Canvas ao plano de negócios pode ser uma alternativa para se obterem, rapidamente, planos de negócios simples e completos.

O plano de negócios é considerado uma ferramenta de gestão do empreendedor, já que o auxilia na tomada de decisões. No passado, esses planos eram desenvolvidos em

detalhes. Devido à necessidade de respostas cada vez mais rápidas, os empreendedores foram otimizando a escrita de seus planos de negócio, de forma que ficassem mais enxutos em quantidade de páginas e mais fáceis de serem atualizados. Atualmente, em poucas páginas (em média, de 15 a 25), é possível se obter um plano de negócios para qualquer empresa. As informações levantadas na análise de oportunidade (modelo de negócio Canvas e pesquisa primária) poderão ser aproveitadas praticamente de maneira integral no desenvolvimento do plano de negócios, em especial na fase de análise do mercado.

2. Análise das alternativas

A – Alternativa incorreta.

JUSTIFICATIVA. Uma startup, usualmente, identifica a oportunidade mais interessante (o que pode ser feito por meio do Canvas) e, depois, parte para o plano de negócios. Isso inclui as startups financeiras.

B – Alternativa correta.

JUSTIFICATIVA. A startup pode utilizar o quadro Canvas para a identificação da oportunidade mais interessante, o que corresponde à validação da ideia de negócio. Posteriormente, pode utilizá-lo, também, como auxílio na elaboração do plano de negócios da empresa.

C – Alternativa incorreta.

JUSTIFICATIVA. Todas as startups, incluindo as de educação, podem se beneficiar do uso do Canvas. Não há garantia de que os modelos de negócios nasçam validados.

D – Alternativa incorreta.

JUSTIFICATIVA. Assim como todas as startups, as startups de logística podem utilizar o Canvas em conjunto com o plano de negócios. Além disso, um produto mínimo viável (MVP) é uma estratégia de desenvolvimento de produto, não um substituto para a estratégia de identificação da oportunidade ou para o plano de negócios.

E – Alternativa incorreta.

JUSTIFICATIVA. Todas as startups, inclusive as de impacto social, precisam considerar seus segmentos de clientes. Mesmo que uma startup de impacto social tenha um segmento de

cliente principal, ela ainda precisa entender esse segmento para poder atendê-lo efetivamente.

3. Indicações bibliográficas

- DORNELES, J. *Plano de negócios com o modelo Canvas: guia prático de avaliação de ideias de negócio*. São Paulo: Empreende, 2020.
- ROSA, J. A.; MARÓSTICA, E. *Modelos de negócios: organização e gestão*. São Paulo: Cengage Learning, 2012.

Questão 5

Questão 5.⁵

Um dos fatores essenciais para o sucesso de um Processo de Software é a habilidade de capturar e gerenciar as mudanças e as oportunidades de negócio. Muitas vezes, as oportunidades de negócio são perdidas em função do tempo requerido para efetivar mudanças nos sistemas de software ou em função de defeitos nesse sistema. Basicamente, o software é o Produto do Processo de Software. Assim, identificar o grau de desenvolvimento do processo de software de uma empresa é o passo inicial para o seu melhoramento. O Modelo de Maturidade da Capacidade (CMMi) descreve práticas, processos e atividades de gestão, guiando o melhoramento do Processo de Software para atingir o mais elevado nível de maturidade da capacidade.

Considerando os diversos aspectos do processo de desenvolvimento de um produto ou sistema de software, avalie as afirmativas.

- I. A maturidade implica o crescimento da capacidade e indica tanto a efetividade do Processo de Software quanto o grau de consistência com que este é aplicado nos diferentes projetos desenvolvidos pela organização.
- II. A maturidade do Processo de Software implica melhorias contínuas do processo com base em etapas evolutivas que se caracterizam pela capacidade de inovação e absorção de novas tecnologias.
- III. A maturidade do Processo de Software implica a sua institucionalização por meio de políticas e padrões organizacionais levando a uma cultura corporativa de métodos, práticas e procedimentos com efetividade plenamente reconhecida.
- IV. Cada nível do CMMi implica a satisfação de um conjunto de metas de processo que estabilizam um componente importante do Processo de Software e representa o seu estágio de maturidade da capacidade.

É correto apenas o que se afirma em

- A. III.
- B. I e II.
- C. II e IV.
- D. I, III e IV.
- E. I, II, III e IV.

⁵Questão 33 – Enade 2021.

1. Análise da questão

A questão requer conhecimentos a respeito de processos de desenvolvimento de software e do Modelo de Maturidade da Capacidade (CMMi). Esses temas serão tratados, de forma introdutória, em sequência.

1.1. Maturidade e capacidade

Na área de engenharia de software, um **processo de desenvolvimento de software** é um conjunto de ações e de atividades relacionadas, realizadas para obtermos um produto de software. Em outras palavras, o processo é uma “receita” que é seguida por um projeto, com o intuito de confeccionar um produto.

No contexto de processo de software, o termo **maturidade** refere-se ao grau de formalidade e de otimização dos processos, ou seja, a maturidade de um processo de software indica o quão bem desenvolvido, gerenciado e refinado é o processo. De forma geral, quanto mais maduro é o processo, mais eficaz e previsível ele se torna.

Analogamente, a maturidade de uma organização mede o grau de competência, técnica e gerencial, que essa organização tem para produzir software de boa qualidade, dentro de prazos e a custos razoáveis e previsíveis. Em organizações com baixa maturidade, os processos de software geralmente podem ser informais, ou seja, não são documentados e existem apenas na mente de seus praticantes.

A existência de processos definidos é necessária para a maturidade das organizações produtoras de software, pois eles permitem que a organização tenha um modo de operação padronizado e reproduzível. Isso facilita a capacitação das pessoas e torna o funcionamento da organização menos dependente de determinados indivíduos. Entretanto, não é suficiente que os processos sejam definidos. Processos rigorosamente definidos, mas não alinhados com os objetivos da organização, podem ser entraves burocráticos, e não fatores de produção.

No contexto de processo de software, o termo **capacidade** refere-se à gama de resultados esperados que podem ser alcançados seguindo um processo. A capacidade de um processo de software pode ser entendida como a eficácia do processo em alcançar seus resultados pretendidos, como a qualidade do produto de software, a eficiência na utilização de recursos e a satisfação do cliente.

Para tornar uma organização mais madura e capacitada, é preciso melhorar a qualidade dos seus processos. O maior critério de êxito dos processos é a medida do quanto eles

contribuem para que os produtos sejam entregues aos clientes com melhor qualidade, por menor custo e em prazo mais curto.

1.2. Níveis de maturidade do CMMi

Diversas organizações do mundo propuseram paradigmas para a melhora dos processos dos setores produtivos. Algumas desenvolveram paradigmas para a melhoria dos processos de software. Entre eles, estão os paradigmas do tipo modelos de capacitação. O modelo de capacitação serve de referência para avaliar a maturidade dos processos de uma organização.

Um importante modelo de capacitação para a área de software é o CMMi (Capability Maturity Model Integration), do Software Engineering Institute (SEI), citado no enunciado da questão.

O CMMi funciona como um modelo de referência, que contém metas e práticas necessárias à maturidade de processos de software. O CMMi apresenta cinco níveis de maturidade, cada um representando um estágio diferente no desenvolvimento da capacidade organizacional. Cada nível de maturidade no modelo CMMi implica a satisfação de um conjunto específico de metas de processo. Esses níveis estão elencados a seguir.

- **Nível 0 (Incompleto):** neste nível, o processo não funciona ou não atinge todas as metas definidas pelo CMMi para a capacidade nível 1.
- **Nível 1 (Executado):** todas as metas específicas das áreas do processo, definidas pelo CMMi, foram satisfeitas. Neste nível, os processos são geralmente pouco controlados. A organização não fornece um ambiente estável e controlado, e os projetos frequentemente ultrapassam o prazo e o orçamento. No entanto, as tarefas necessárias para produzir os produtos definidos estão sendo executadas.
- **Nível 2 (Controlado):** os projetos deste nível têm processos gerenciados, além de serem planejados, executados, medidos e controlados. Organizações que se encontram no nível 2 de maturidade ainda carregam processos iniciais, mas com o diferencial de que controlam com mais facilidade os custos e os prazos dos projetos.
- **Nível 3 (Definido):** no nível 3, os processos são bem caracterizados e compreendidos e são descritos em padrões, procedimentos, ferramentas e métodos. Empresas que se encontram no terceiro nível já têm processos bem definidos, pois suas documentações são padronizadas e suas práticas são uniformes. Desse modo, todos sabem o que deve ser feito em cada etapa do desenvolvimento do produto.

- **Nível 4 (Controlado quantitativamente):** as organizações que se encontram no nível 4 já são consideradas de alta maturidade. Elas usam análise estatística quantitativa para determinar, identificar e gerenciar a estabilidade e a capacidade do processo e seus impactos nos objetivos de obtenção de qualidade e de desempenho.
- **Nível 5 (Otimizado):** o nível de maturidade 5, obviamente, também é um nível de alta maturidade organizacional. Organizações de nível 5, além do controle estatístico, focam em melhoria contínua, com o intuito de alcançar processos flexíveis, capazes de responder às oportunidades e às mudanças. Além disso, essas organizações buscam estabilizar os seus processos, aumentando a previsibilidade de resultados e, assim, criando um ambiente mais adequado para absorver inovações.

2. Análise das afirmativas

I – Afirmativa correta.

JUSTIFICATIVA. A maturidade de um processo implica o crescimento da sua capacidade já que, à medida que um processo amadurece, ele se torna mais capaz de atingir seus resultados pretendidos.

II – Afirmativa incorreta.

JUSTIFICATIVA. As etapas evolutivas propostas pelo modelo CMMi não se caracterizam, necessariamente, pela capacidade de inovação e de absorção de novas tecnologias. Embora esses sejam fatores importantes para o desenvolvimento e a melhora de processos, a maturidade do processo de software está mais diretamente relacionada à previsibilidade, à eficácia e ao controle do processo. A capacidade de absorção de novas tecnologias é mais diretamente associada ao nível 5 do modelo CMMi.

III – Afirmativa correta.

JUSTIFICATIVA. A maturidade de um processo de software está diretamente relacionada à sua institucionalização. Conforme um processo de software amadurece, ele torna-se parte integrante da organização, sendo apoiado por políticas e por padrões organizacionais. Essas políticas e padrões ajudam a garantir que o processo seja aplicado de maneira consistente em toda a organização, o que leva a resultados mais previsíveis e de alta qualidade.

IV – Afirmativa correta.

JUSTIFICATIVA. Cada nível de maturidade do modelo CMMi representa um estágio diferente na maturidade do processo de software. À medida que uma organização avança nos níveis, ela demonstra capacidades crescentes de gerenciar e de controlar seus processos. Cada nível de maturidade no modelo CMMi realmente implica a satisfação de um conjunto específico de metas de processo. Essas metas são essenciais para atingir um objetivo do processo de software.

Alternativa correta: D.

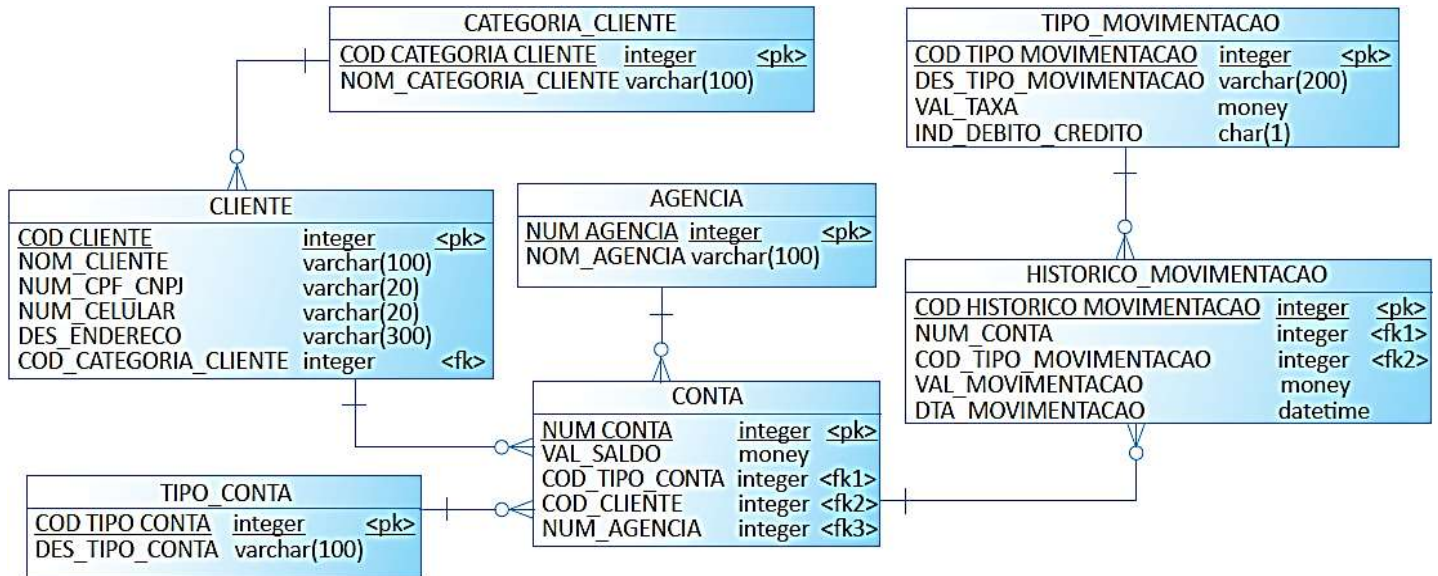
3. Indicações bibliográficas

- MONTONI, M. *Níveis de maturidade do CMMI: Entenda!* Disponível em <<https://promovesolucoes.com/niveis-maturidade-cmmi/>>. Acesso em 04 mar. 2024.
- PAULA FILHO, W. de P. *Engenharia de software: produtos*. Rio de Janeiro: LTC, 2019.
- PRESSMAN, R. S.; MAXIM, B. R. *Engenharia de software: uma abordagem profissional*. Porto Alegre: AMGH, 2021.

Questão 6

Questão 6.⁶

O modelo de dados relacional a seguir representa de forma simplificada o banco de dados para um sistema de informação bancário.



Usando a linguagem SQL (Structured Query Language), indique a consulta que recupera os nomes dos clientes e o valor médio movimentado por eles em sua conta até o momento. Primeiro devem ser exibidos os clientes com maior média de valor movimentado.

A. `SELECT CL.NOME, HM.VAL_MOVIMENTADO`

`FROM CLIENTE CL INNER JOIN CONTA CO ON CL.COD_CLIENTE = CO.COD_CLIENTE`
`INNER JOIN HISTORICO_MOVIMENTACAO HM ON CO.NUM_CONTA = HM.NUM_CONTA`
`ORDER BY HM.VAL_MOVIMENTADO`

B. `SELECT CL.NOME, AVG(HM.VAL_MOVIMENTADO)`

`FROM CLIENTE CL INNER JOIN CONTA CO ON CL.COD_CLIENTE = CO.COD_CLIENTE`
`INNER JOIN HISTORICO_MOVIMENTACAO HM ON CO.NUM_CONTA = HM.NUM_CONTA`
`ORDER BY HM.VAL_MOVIMENTADO DESC`

C. `SELECT CL.NOME, AVG(HM.VAL_MOVIMENTADO)`

`FROM CLIENTE CL INNER JOIN CONTA CO ON CL.COD_CLIENTE = CO.COD_CLIENTE`
`INNER JOIN HISTORICO_MOVIMENTACAO HM ON CO.NUM_CONTA = HM.NUM_CONTA`
`GROUP BY CL.COD_CLIENTE, CL.NOME`
`ORDER BY AVG(HM.VAL_MOVIMENTADO) DESC`

⁶Questão 17 – Enade 2021.

- D. `SELECT CL.NOME, AVG(HM.VAL_MOVIMENTADO)`
`FROM CLIENTE CL LEFT OUTER JOIN CONTA CO ON CL.COD_CLIENTE =`
`CO.NUM_CONTA`
`LMEOFVIT MEONUTTAECR AJO OIN HISTORICO_MOVIMENTACAO HM ON CO.NUM_CONTA`
`= HM.COD_HISTORICO_`
`GROUP BY CL.NOME`
`ORDER BY AVG(HM.VAL_MOVIMENTADO) DESC`
- E. `SELECT CL.NOME, AVG(HM.VAL_MOVIMENTADO)`
`FROM CLIENTE CL LEFT OUTER JOIN CONTA CO ON CL.COD_CLIENTE =`
`CO.COD_CLIENTE`
`LEFT OUTER JOIN HISTORICO_MOVIMENTACAO HM ON CO.NUM_CONTA =`
`HM.NUM_CONTA`
`GROUP BY CL.NOME`
`ORDER BY AVG(HM.VAL_MOVIMENTADO) DESC`

1. Análise da questão

A questão requer conhecimentos sólidos a respeito da linguagem SQL (Structured Query Language), além de exigir a interpretação de um diagrama entidade-relacionamento. Esses temas serão abordados brevemente a seguir.

1.1. Banco de dados relacional

De forma geral, podemos pensar em um **banco de dados** como uma coleção de dados estruturados, organizados e armazenados de forma persistente. Nesse contexto, um **Sistema de Gerência de Banco de Dados** (SGBD) é um software que fornece a interface entre os dados que estão armazenados no banco e o usuário.

Um **Sistema de Gerência de Banco de Dados Relacional** (SGBDR) é um tipo de SGBD que armazena dados em uma estrutura de banco de dados relacional. Isso significa que os dados são organizados em tabelas, que se relacionam umas com as outras. A estrutura de um banco de dados relacional pode ser descrita visualmente por um **diagrama entidade-relacionamento**, que é um diagrama que utiliza simbologia específica para demonstrar relacionamentos entre entidades. Vamos, a seguir, estudar o básico a respeito dessa modelagem.

Em um banco de dados relacional, uma **entidade** é um objeto ou um conceito sobre o qual queremos armazenar informações. Cada entidade é representada como uma tabela no banco de dados. Por exemplo, no diagrama que vemos no enunciado da questão, CLIENTE, TIPO_CONTA, AGENCIA, CONTA e HISTORICO_MOVIMENTACAO, CATEGORIA_CLIENTE e TIPO_MOVIMENTACAO representam entidades.

A linha que une duas entidades estabelece uma **relação** entre elas. Considere a relação entre as entidades CATEGORIA_CLIENTE e CLIENTE reproduzida na figura 1. O símbolo da “bolinha” indica “zero”, e o símbolo do “pé de galinha” significa “muitos”. Quando vemos o símbolo que une uma “bolinha e um pé de galinha” colocado no lado da entidade CLIENTE, isso significa que uma mesma categoria pode ser usada por zero ou por muitos clientes. Já o “tracinho”, visto no lado da entidade CATEGORIA_CLIENTE, significa “um”. Ou seja, ele indica que, a cada cliente, está associada uma categoria.

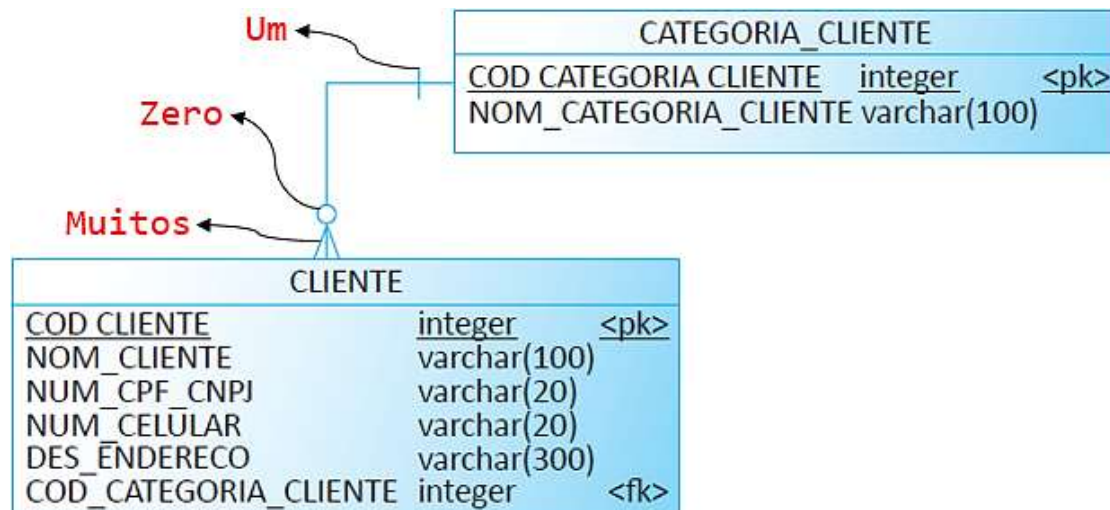


Figura 1. Trecho do modelo de banco de dados relacional.

ENADE Tecnologia em Análise e Desenvolvimento de Sistemas 2021. Disponível em <<https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enade/provas-e-gabaritos>>. Acesso em 11 mar. 2024 (com adaptações).

Depois de identificadas as entidades e estabelecidas as suas respectivas relações, é necessário indicar que **atributos** vão fazer parte de cada uma delas, isto é, que dados fazem parte da definição da entidade CATEGORIA_CLIENTE e da entidade CLIENTE. Cada atributo representa uma coluna da tabela (entidade). Ambas as entidades são vistas a seguir, com seus respectivos atributos listados entre parênteses.

```
CATEGORIA_CLIENTE(COD_CATEGORIA_CLIENTE, NOM_CATEGORIA_CLIENTE)
CLIENTE(COD_CLIENTE, NOM_CLIENTE, NUM_CPF_CNPJ, NUM_CELULAR, DES_ENDERECO,
COD_CATEGORIA_CLIENTE).
```

Os atributos existentes na definição da entidade não têm todos o mesmo grau de importância. Alguns conseguem identificar de forma exclusiva cada uma das ocorrências da

entidade e são escolhidos como **chaves primárias** (em inglês, *primary keys*), que atuam como identificadores. É comum representar as chaves primárias no modelo de dados em negrito ou sublinhadas, para realçá-las, conforme exposto a seguir.

```
CATEGORIA_CLIENTE(COD_CATEGORIA_CLIENTE, NOM_CATEGORIA_CLIENTE)
CLIENTE(COD_CLIENTE, NOM_CLIENTE, NUM_CPF_CNPJ, NUM_CELULAR, DES_ENDERECO,
COD_CATEGORIA_CLIENTE).
```

Note que, para estabelecermos o relacionamento que permita armazenar informações sobre que categoria é utilizada por cada cliente, é necessário colocar na entidade CLIENTE algo que indique a que categoria ele pertence.

É o atributo COD_CATEGORIA_CLIENTE, que colocamos na entidade CLIENTE, que irá indicar que categoria está associada a cada um dos clientes existentes. Perceba, no entanto, que estamos indicando apenas o código da categoria, e não os dados completos dessa categoria. Por quê? Ao optarmos por colocar o identificador (chave primária) da entidade CATEGORIA_CLIENTE na entidade CLIENTE, imediatamente ficamos com uma ligação entre as duas entidades e, a qualquer momento, podemos saber que código e que nome de categoria é associada a cada cliente.

Com esse método, colocamos, na entidade CLIENTE, um atributo extra, denominado **chave estrangeira** (em inglês, *foreign key*), que estabelece uma ligação entre a entidade CATEGORIA_CLIENTE e a entidade CLIENTE.

Depois de termos especificado o modelo de dados, já é possível fazer a implementação física do banco de dados, de acordo com a estrutura especificada no modelo de entidades e relacionamentos. Para isso, teremos que escolher um SGBDR existente no mercado e, então, criar um banco de dados para dar suporte a essas informações.

A criação de um banco de dados pode ser feita por meio de ferramentas que alguns sistemas de gerência disponibilizam para facilitar a execução desse procedimento por parte dos usuários, ou por meio de um comando CREATE DATABASE. Esse comando é um dos muitos que fazem parte da linguagem SQL, que será abordada a seguir.

1.2. Algumas cláusulas SQL

A **linguagem SQL** (Structured Query Language) é uma linguagem de programação voltada ao acesso e à manipulação de dados em um banco de dados relacional. Atualmente, a linguagem SQL é considerada um padrão dos SGBDRs e, por conta disso, praticamente todos os fabricantes a integram em seus produtos.

Uma instrução SQL é formada por componentes denominados **cláusulas**. Cada cláusula executa uma tarefa para a instrução SQL. Para executarmos uma instrução de seleção de dados, por exemplo, algumas cláusulas são necessárias. Algumas das cláusulas e funções SQL, presentes no texto das alternativas da questão, são brevemente descritas a seguir.

- SELECT: usada para selecionar atributos de interesse de uma ou de mais tabelas (entidades).
- FROM: usada para especificar as tabelas das quais os atributos listados na cláusula SELECT serão selecionados.
- INNER JOIN: usada para combinar linhas de duas ou de mais tabelas com base em uma coluna (atributo) relacionada entre elas. Representa a interseção entre os dados dessas tabelas.
- ON: usada para especificar a condição de junção ao usar o comando JOIN.
- GROUP BY: usada para agrupar e organizar o resultado das funções de agregação. Pode ser aplicada caso seja necessário reunir conjuntos de informações.
- ORDER BY: usada para classificar os resultados, em ordem crescente ou decrescente, com base em uma ou mais colunas.
- DESC: termo usado com ORDER BY para classificar os resultados em ordem decrescente.
- AVG: função que retorna a média de todos os valores em uma coluna específica.

Por exemplo, para consultar o atributo nome da tabela alunos, a instrução a seguir, que utiliza duas cláusulas, pode ser utilizada.

```
SELECT nome  
FROM alunos;
```

Para consultar o atributo nome da tabela alunos, em que os nomes devem aparecer classificados em ordem decrescente em relação ao atributo nota, podemos usar a instrução a seguir. Nesse caso, o nome do aluno com maior nota aparecerá primeiro no resultado da consulta, e o nome do aluno com menor nota aparecerá por último.

```
SELECT nome  
FROM alunos  
ORDER BY nota DESC;
```

Considere uma entidade, denominada pilotos, que contém os dados de pilotos da Fórmula 1. Para mostrar a pontuação total, por país, dos pilotos de Fórmula 1, podemos utilizar a instrução a seguir.

```
SELECT SUM(pontos) AS SomaDosPontos, pais AS Pais
FROM pilotos
GROUP BY pais
ORDER BY SomaDosPontos;
```

Em primeiro lugar, somamos os pontos por meio de SUM(pontos). No entanto, ao utilizar esse trecho, teremos o total de pontos dos pilotos de todos os países. Como queremos a soma por país, utilizamos a cláusula GROUP BY, que possibilita somar os pontos dos pilotos de cada país. Para visualizarmos melhor os resultados, ordenamos pelo resultado da soma dos pontos. O resultado dessa busca pode ser visto na figura 2.

Além disso, adotamos apelidos (alias) para as colunas de interesse. Esses apelidos foram posicionados após o termo AS. Esse recurso pode ser utilizado para facilitar a visualização e a compreensão dos dados com os quais estamos trabalhando.

SomaDosPontos	Pais
0	Australia
0	India
0	Finlandia
8	Suiça
21	Italia
27	Russia
32	Japão
136	Polonia
191	Brasil
242	Austria
263	Espanha
454	Inglaterra
545	Alemanha

Figura 2. Resultado da consulta da pontuação total, por país, dos pilotos de Fórmula 1. CARDOSO, V.; CARDOSO, G. *Linguagem SQL: fundamentos e práticas*. São Paulo: Saraiva, 2013.

2. Resolução da questão

No contexto da questão, CL é um alias dado à tabela CLIENTE, CO é um alias para a tabela CONTA e HM é um alias para a tabela HISTORICO_MOVIMENTACAO. Os alias foram inferidos dos próprios comandos presentes nas alternativas da questão. Eles não podem ser inferidos diretamente do diagrama, pois ele mostra apenas as tabelas, suas colunas e as relações entre elas.

Com base nisso, a consulta que recupera o nome dos clientes e o valor médio movimentado por eles até o momento, exibindo primeiro os clientes com maior média de valor movimentado, é disposta a seguir.

```
1. SELECT CL.NOME, AVG(HM.VAL_MOVIMENTADO)
2. FROM CLIENTE CL INNER JOIN CONTA CO ON CL.COD_CLIENTE = CO.COD_CLIENTE
3. INNER JOIN HISTORICO_MOVIMENTACAO HM ON CO.NUM_CONTA = HM.NUM_CONTA
4. GROUP BY CL.COD_CLIENTE, CL.NOME
5. ORDER BY AVG(HM.VAL_MOVIMENTADO) DESC
```

Essa consulta junta as tabelas CLIENTE, CONTA e HISTORICO_MOVIMENTACAO com base em suas chaves primárias e estrangeiras correspondentes. Em seguida, agrupa os resultados por COD_CLIENTE e NOME (ambos na tabela CLIENTE) e calcula a média do VAL_MOVIMENTADO (na tabela HISTORICO_MOVIMENTACAO) para cada cliente. Por fim, ordena os resultados pela média do VAL_MOVIMENTADO, em ordem decrescente. As etapas da consulta são explicadas linha a linha, a seguir.

Linha 1. A cláusula SELECT seleciona duas colunas para exibição nos resultados: NOME (da tabela CLIENTE) e média aritmética (AVG) de VAL_MOVIMENTADO (da tabela HISTORICO_MOVIMENTACAO).

Linha 2. Essa linha junta a tabela CLIENTE com a tabela CONTA, sendo que o COD_CLIENTE na tabela CLIENTE é igual ao COD_CLIENTE na tabela CONTA. Isso significa que estamos combinando registros das duas tabelas que pertencem ao mesmo cliente.

Linha 3. Aqui, juntamos a tabela resultante da junção anterior (linha 2) com a tabela HISTORICO_MOVIMENTACAO (alias HM), sendo que o NUM_CONTA na tabela CONTA é igual ao NUM_CONTA na tabela HISTORICO_MOVIMENTACAO. Nesse caso, estamos combinando registros que pertencem à mesma conta.

Linha 4. Essa linha agrupa os resultados pelas colunas COD_CLIENTE e NOME, da tabela CLIENTE. Isso significa que, para cada cliente único (identificado por COD_CLIENTE e NOME), todas as suas linhas serão agrupadas em uma única linha nos resultados. É utilizada a chave primária (COD_CLIENTE) no agrupamento, o que é importante no caso de haver clientes homônimos.

Linha 5. Esse trecho ordena os resultados pela média (AVG) da coluna VAL_MOVIMENTADO da tabela HISTORICO_MOVIMENTACAO, em ordem decrescente (DESC). Isso faz com que os clientes com maior média de valor movimentado sejam exibidos primeiro nos resultados.

Alternativa correta: C.

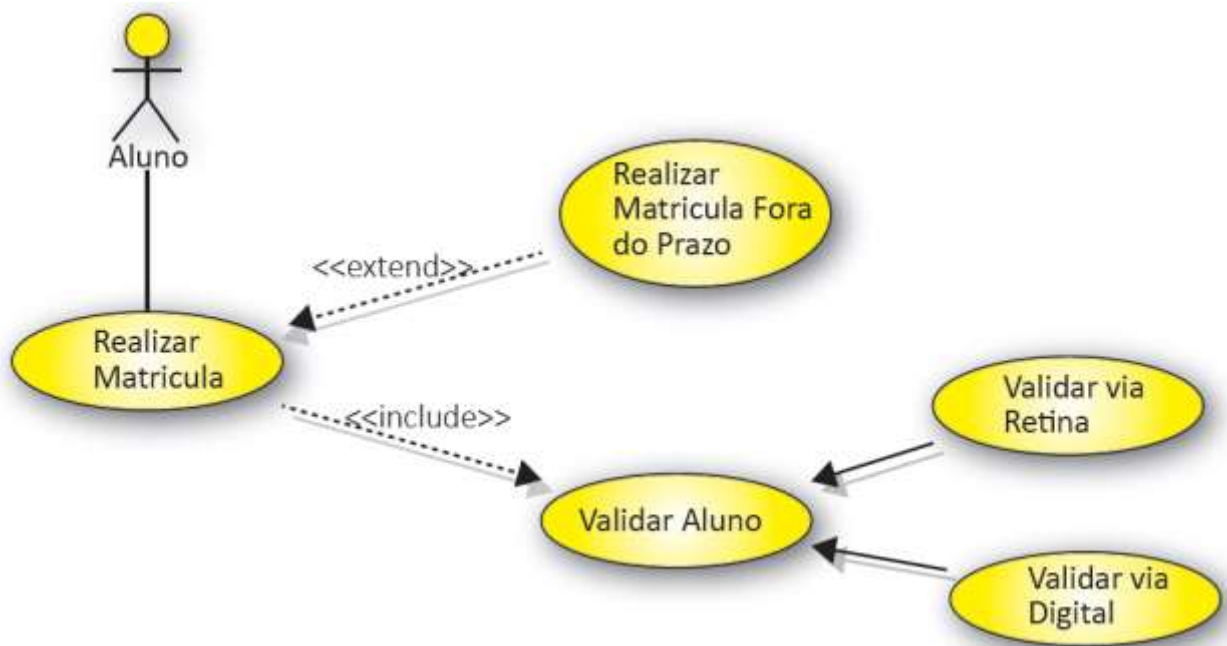
3. Indicações bibliográficas

- CARDOSO, V.; CARDOSO, G. *Linguagem SQL: fundamentos e práticas*. São Paulo: Saraiva, 2013.
- DAMAS, L. *SQL - Structured Query Language*. Rio de Janeiro: LTC, 2014.
- MICROSOFT. *Acessar SQL: conceitos básicos, vocabulário e sintaxe*. Disponível em <<https://support.microsoft.com/pt-br/topic/acessar-sql-conceitos-b%C3%A1sicos-vocabul%C3%A1rio-e-sintaxe-444d0303-cde1-424e-9a74-e8dc3e460671#:~:text=Cl%C3%A1usulas%20SQL,as%20cl%C3%A1usulas%20SQL%20mais%20comuns->>>. Acesso em 25 mar. 2025.

Questão 7

Questão 7.⁷

Foi solicitado o desenvolvimento de um software para o controle e a gestão das matrículas dos alunos em uma universidade. Os alunos poderão realizá-las diretamente no sistema, em um terminal a ser disponibilizado na universidade. Além dessas características, o sistema deverá considerar as restrições representadas no diagrama a seguir.



Com base no texto e no diagrama, avalie as afirmativas.

- I. O aluno poderá realizar a matrícula fora do prazo e as tratativas são as mesmas da matrícula feita dentro do prazo.
- II. O aluno poderá, opcionalmente, realizar a matrícula fora do prazo.
- III. O sistema, obrigatoriamente, deverá validar o aluno que realizar a matrícula fora do prazo.
- IV. A validação do aluno deverá ser feita a partir da sua matrícula informada no sistema.
- V. A validação do aluno, a ser feita por retina, é diferente da validação feita pela digital.

É correto apenas o que se afirma em

- A. I e IV.
- B. I, II e IV.
- C. I, III e V.
- D. II, III e V.
- E. II, III, IV e V.

⁷Questão 31 – Enade 2021.

1. Análise da questão

A questão cobra conhecimentos a respeito de diagramas de caso de uso, um dos tipos de diagramas UML (Unified Modeling Language). A seguir, faremos uma breve introdução ao tema e, em sequência, analisaremos o diagrama presente no enunciado da questão.

1.1. Diagramas de caso de uso

Os **casos de uso** são uma técnica utilizada para levantar os requisitos funcionais de um sistema. Eles servem para descrever as interações típicas entre os usuários de um sistema e o próprio sistema, oferecendo uma narrativa sobre como o sistema é utilizado.

Os casos de uso podem ser modelados por diagramas UML, denominados **diagramas de casos de uso**. Esses diagramas têm o objetivo de descrever, visualmente, como os usuários interagem com um sistema.

Nesse contexto, os usuários são referidos como **atores**. Os atores realizam os casos de uso, que são as possíveis ações que podem ser tomadas pelos atores. Cada caso de uso tem um ator principal, que pede ao sistema a execução de um serviço.

Comumente, o diagrama de caso de uso não oferece muitos detalhes, mas apresenta uma visão geral do relacionamento entre casos de uso, atores e sistemas. Por isso, é comum usar esses diagramas para complementar um caso de uso descrito em texto.

Em relação à simbologia adotada em um diagrama, o caso de uso é representado por uma forma oval rotulada. Bonecos palito representam os atores, e a participação do ator no sistema é modelada com uma linha entre o(s) ator(es) e o caso de uso.

Os conceitos de <<include>> e <<extend>> designam relações entre os casos de uso. A relação <<include>> indica que um caso de uso, para ter sua funcionalidade executada, precisa chamar outro caso de uso. A relação <<extend>> significa que o caso de uso estendido vai funcionar exatamente como o caso de uso base, porém com alguns passos novos inseridos no caso de uso estendido.

1.2. Análise do diagrama do enunciado

O diagrama de caso de uso apresentado no enunciado da questão traz o aluno como ator e os seguintes casos de uso: Realizar matrícula, Realizar matrícula fora do prazo, Validar aluno, Validar via retina, Validar via digital. Esses casos de uso são detalhados a seguir.

- Realizar Matrícula: essa é a primeira ação que o aluno pode tomar. Ele pode matricular-se diretamente no sistema.
- Realizar Matrícula Fora do Prazo: esse é um caso de uso estendido do passo anterior, “Realizar matrícula”. Isso significa que o aluno tem a opção de matricular-se após o prazo estipulado, possivelmente sob condições específicas.
- Validar Aluno: esse é um passo obrigatório após a matrícula (dentro ou fora do prazo). O sistema deve validar o aluno que se matriculou. O diagrama indica que “Realizar matrícula” inclui “Validar aluno”. Isso é indicado pela seta de <<include>> que conecta “Realizar matrícula” a “Validar aluno”.
- Validar via Retina e Validar via Digital: esses são dois métodos distintos de validação do aluno. Eles são diferentes entre si, mas ambos são utilizados para confirmar a identidade do aluno.

2. Análise das afirmativas

I – Afirmativa incorreta.

JUSTIFICATIVA. O diagrama mostra que o aluno pode realizar a matrícula fora do prazo, mas não especifica que as tratativas são as mesmas da matrícula feita dentro do prazo. Nesse contexto, o caso de uso “Realizar matrícula fora do prazo” é uma extensão do caso de uso “Realizar matrícula”. Esperamos, portanto, que “Realizar matrícula fora do prazo” funcione como “Realizar matrícula”, mas com a inserção de alguns passos específicos.

II – Afirmativa correta.

JUSTIFICATIVA. O diagrama mostra que o aluno tem a opção de realizar a matrícula fora do prazo, já que esse é um dos casos ilustrados.

III – Afirmativa correta.

JUSTIFICATIVA. O diagrama mostra que, independentemente de quando a matrícula é realizada (dentro ou fora do prazo), o aluno deve ser validado.

IV – Afirmativa incorreta.

JUSTIFICATIVA. O diagrama não especifica que a validação do aluno é feita a partir da matrícula informada no sistema. Ele mostra que a validação é feita via retina ou via digital.

V – Afirmativa correta.

JUSTIFICATIVA. O diagrama mostra, claramente, que existem dois métodos diferentes de validação: via retina e via digital. Cada um desses métodos representa um caso de uso no diagrama.

Alternativa correta: D.

3. Indicações bibliográficas

- FOWLER, M. *UML essencial: um breve guia para a linguagem-padrão de modelagem de objetos*. Porto Alegre: Bookman, 2005.
- LEE, R. C.; TEPFENHART, W. M. *UML e C++: guia prático de desenvolvimento orientado a objeto*. São Paulo: Makron Books, 2001.

Questão 8

Questão 8.⁸

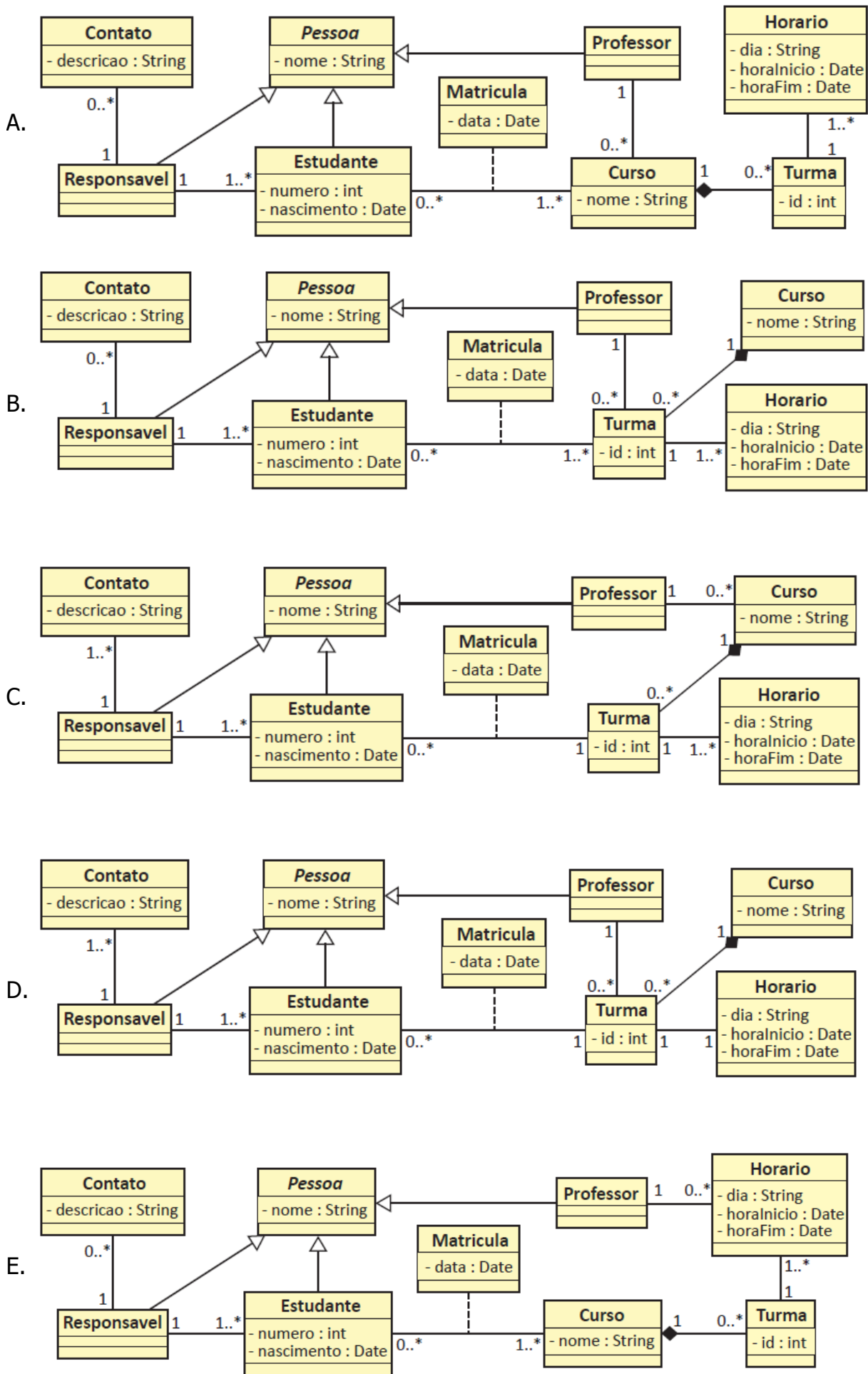
Uma fundação municipal de arte e cultura oferece cursos para crianças e adolescentes de até 16 anos. Os registros de matrículas são realizados em uma planilha eletrônica ilustrada a seguir, o que dificulta bastante o controle dos dados frente ao número expressivo de estudantes, muitos dos quais solicitam matrícula em mais de um curso. A administração da fundação, então, solicitou o desenvolvimento de um programa para facilitar o registro e a consulta desses dados, incluindo-se a data de efetivação de cada matrícula, para fins de controles específicos da secretaria.

Nº	NOME ESTUDANTE	DATA NASC.	NOME RESPONSÁVEL	CONTATOS	CURSO	HORÁRIO	PROFESSOR(A)
10	Ada Lovelace	10/12/2009	Charles Babbage	(99)99999-0101 charles@babbage.com	Violão Básico Desenho	9h–10h (ter–qui) 10h–11h (seg–qua)	Dennis Ritchie John Backus
11	Ole-Johan Dahl	12/10/2008	Kristen Nygaard	-	Desenho	9h–10h (seg–qua)	Mary Keller
12	Grace Hopper	09/12/2010	Howard Aiken	(99)99999-0091 (99)99999-0095	Balé	8–10h (sex)	Hedy Lamarr
13	Alan Turing	23/06/2009	Joan Clarke	(99)99999-0231	Desenho Violão Básico	9h–10h (seg–qua) 9h–10h (ter–qui)	Mary Keller Dennis Ritchie
14	Dorothy Vaughan	20/09/2009	Katherine Johnson	(99)99999-0201	Balé	8h–10h (sex)	Hedy Lamarr
...	...	...	...	...	...	...	...

A equipe de desenvolvimento de softwares da prefeitura, após uma breve entrevista com a administração da fundação e de posse da planilha eletrônica, modelou um Diagrama de Classes como parte da especificação dos requisitos do sistema.

Considerando o cenário descrito, assinale a opção a seguir que exhibe o Diagrama de Classes modelado corretamente pela equipe de desenvolvimento.

⁸Questão 15 – Enade 2021.



1. Análise da questão

A questão cobra conhecimentos a respeito de diagramas de classes, um dos tipos de diagramas UML (Unified Modeling Language). A seguir, faremos uma breve introdução ao tema e, em sequência, analisaremos a tabela disposta no enunciado da questão.

1.1. Diagramas de classes

No contexto dos diagramas UML (Unified Modeling Language), os **diagramas de classe** são aqueles capazes de modelar as classes de um programa computacional desenvolvido em uma linguagem de programação orientada a objetos. Esses diagramas exibem as classes, seus respectivos atributos, suas operações e seus relacionamentos com outras classes do sistema.

Na figura 1, vemos um diagrama de classes simples, que modela um programa que trabalha com processamentos de pedidos. Os retângulos do diagrama representam as classes, que estão divididas em três compartimentos: o nome da classe (em negrito), seus atributos e suas operações. Na mesma figura, também vemos dois tipos de relacionamentos entre as classes: associações e generalizações.

Os **atributos** descrevem uma propriedade com uma linha de texto dentro do retângulo da classe em si. A sintaxe completa de um atributo é exposta a seguir.

```
visibilidade nome: tipo multiplicidade = valor_por_omissão {lista de propriedades}
```

O marcador de visibilidade indica se o atributo é público (+) ou privado (-), entre outras opções de visibilidade. O nome do atributo indica como a classe se refere ao atributo, sendo o nome um campo obrigatório. O tipo do atributo indica uma restrição sobre o tipo de objeto que podemos colocar no atributo. A multiplicidade é uma indicação de quantos objetos podem preencher a propriedade. O valor_por_omissão é o valor dado ao objeto, caso isso não seja especificado durante sua criação. A {lista de propriedades} permite que indiquemos propriedades adicionais para o atributo.

multiplicidade 0..* indica mínimo de 0 e máximo ilimitado de objetos (o que pode ser resumido apenas com *).

1.1.1. Relacionamentos

A **associação** é um relacionamento entre classes que indica que, de algum modo, as instâncias de uma classe estão ligadas às instâncias de outra classe. Uma **associação unidirecional**, representada por uma linha cheia entre duas classes com uma ponta de seta, é um tipo de relacionamento entre classes que indica uma classe de origem e uma classe de destino. O nome da propriedade fica no destino da associação, com sua multiplicidade.

Tanto atributos quanto associações podem descrever propriedades. De forma geral, tendemos a utilizar atributos para coisas “pequenas”, como datas ou valores booleanos, e associações para classes mais significativas, como clientes e pedidos.

Podemos ter, também, **associações bidirecionais**, representadas por uma linha cheia entre duas classes com uma ponta de seta em cada extremidade, conforme exposto na figura 2 (alternativamente, essa associação pode ser representada apenas por uma linha cheia). Nesse caso, temos um par de propriedades vinculadas. No exemplo, a classe Carro tem a propriedade proprietário:Pessoa[0..1] e a classe Pessoa tem a propriedade carros:Carro[*]. O vínculo entre elas indica que, se seguirmos as duas propriedades, nós devemos retornar a um conjunto que contém seu ponto de partida. Por exemplo, se você começar com um veículo MG Midget particular, encontrar seu proprietário e depois ver os carros desse proprietário, esse conjunto deverá conter o MG Midget do qual você partiu.

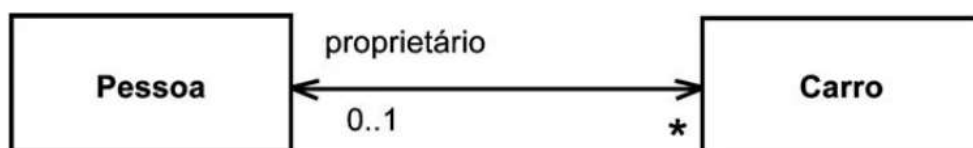


Figura 2. Exemplo de associação bidirecional.
FOWLER, 2005.

Uma **classe de associação** é uma classe que faz parte de um relacionamento de associação entre outras classes. Podemos conectar uma classe de associação a um relacionamento de associação para fornecer informações adicionais sobre o relacionamento. Geralmente, é usada quando ambas as extremidades da associação têm multiplicidade *. Sua utilização permite simplificar esse relacionamento complexo.

Existem casos especiais de associação: a agregação e a composição. A **agregação** representa uma relação todo-parte entre classes. Nessa relação, a perda do “todo” não resulta,

obrigatoriamente, na perda das “partes”. Sua simbologia é uma linha com um losango vazado na classe que corresponde ao todo. Na figura 3, temos a classe Time, que representa a classe “todo”, e Estadio, que representa a classe “parte”. Isso indica que o estádio está agregado ao time, mas ambos podem continuar existindo independentemente.

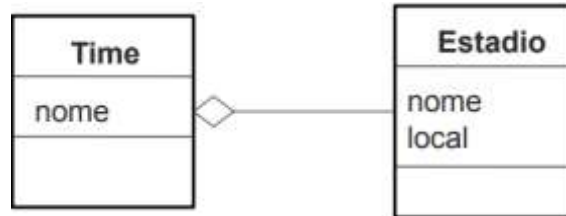


Figura 3. Exemplo de agregação, um tipo de associação entre classes.

Disponível em <https://homepages.dcc.ufmg.br/~figueiredo/disciplinas/aulas/uml-diagrama-classes-relacionamentos_v01>. Acesso em 19 abr. 2025.

Já a **composição** é um tipo de associação que também representa uma relação todo-parte entre classes. Porém a composição representa um forte relacionamento, em que as partes não existem sem o todo, ou o todo não faz sentido sem as partes. Sua simbologia é uma linha com um losango preto na classe que corresponde ao todo. Na figura 4, temos a classe Time, que representa a classe “todo”, e Jogador, que representa a classe “parte”. Isso indica que os jogadores compõem o time, e que o time não “faz sentido” sem seus jogadores.



Figura 4. Exemplo de composição, um tipo de associação entre classes.

Disponível em <https://homepages.dcc.ufmg.br/~figueiredo/disciplinas/aulas/uml-diagrama-classes-relacionamentos_v01>. Acesso em 19 abr. 2025.

As **generalizações**, relacionamentos entre classes representados por uma ponta triangular, indicam que há herança entre classes. Nesse caso, a classe-pai (cujas características são herdadas pela classe-filha) é “apontada” pelo triângulo. No diagrama da figura 1, as classes Cliente Corporativo e Cliente Pessoal, que atuam como classes-filhas, herdam as características de Cliente, que é a classe-pai.

Na figura 5, vemos mais um exemplo de generalização. No caso, a classe Pessoa é a classe-pai, enquanto as classes Professor e Aluno são classes-filhas. Isso significa, por regra geral, que professores e alunos têm suas especificidades, mas que ambos são pessoas.

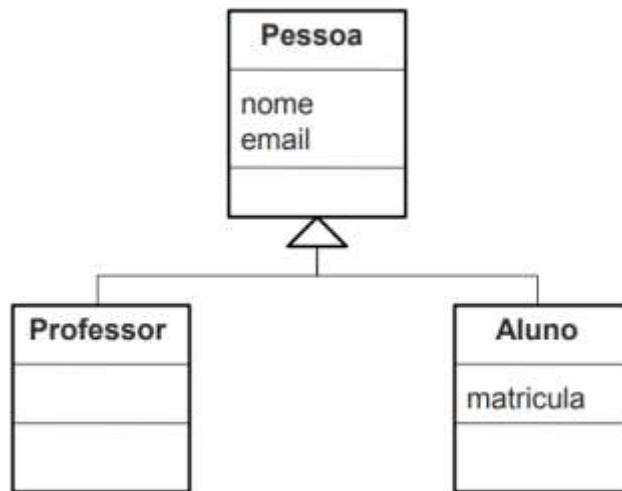


Figura 5. Exemplo de generalização, na qual existe herança entre classes.

Disponível em <https://homepages.dcc.ufmg.br/~figueiredo/disciplinas/aulas/uml-diagrama-classes-relacionamentos_v01>. Acesso em 19 abr. 2025.

2. Análise da tabela do enunciado

Analisando a tabela fornecida e a situação descrita no enunciado da questão, podemos fazer algumas inferências, elencadas a seguir.

- Há três tipos de pessoas presentes na tabela: os estudantes, os professores e os contatos dos estudantes. Logo, esperamos que as classes Estudante, Professor e Responsavel herdem as características da classe Pessoa.
- Cada estudante tem apenas um responsável. Desse modo, a multiplicidade esperada para a associação entre Estudante e Responsavel, na extremidade Responsavel, é 1.
- Cada responsável pode fornecer um ou mais contatos, mas eles não são obrigados a isso. Kristen Nygaard não forneceu qualquer contato; Joan Clarke e Katherine Johnson forneceram apenas um contato; e Charles Babbage e Howard Aiken forneceram dois contatos. Logo, a multiplicidade esperada para a associação entre Responsavel e Contato, na extremidade Contato, é 0..*.
- Cada professor é uma pessoa associada a uma turma, não necessariamente a um curso. Por exemplo, Mary Keller dá aula de Desenho para a turma cujo horário é das 9h às 10h (seg - qua), enquanto John Backus dá aula de Desenho para a turma cujo horário é das 10h às 11h (seg - qua).
- Cada curso pode ter mais de uma turma, conforme exposto no exemplo anterior. No entanto, para um curso ser criado, ele não necessariamente precisa fechar turmas. Logo, a associação entre as classes Curso e Turma deve ter multiplicidade 0..* na extremidade Turma.

- Só faz sentido uma turma existir se ela compuser um curso. Desse modo, esperamos que a classe Turma seja uma “parte” do “todo” Curso.
- Cada estudante pode se matricular para uma ou mais turmas. Assim, a multiplicidade esperada para a associação entre as classes Estudante e Turma, na extremidade Turma, é 1..*.
- Cada turma pode ter suas aulas ministradas em apenas um horário semanal, como é o caso das turmas do curso de Balé, ou pode ter suas aulas divididas em mais dias, como é o caso das turmas dos outros cursos. Logo, a multiplicidade esperada para a associação entre as classes Turma e Horario, na extremidade Horario, é 1..*.

A partir desses requisitos, vamos analisar os diagramas de classe presentes em cada uma das alternativas, a seguir.

3. Análise das alternativas

A – Alternativa incorreta.

JUSTIFICATIVA. O diagrama de classes associa um professor aos cursos, e não um professor às turmas. Desse modo, cada curso poderia ter apenas um professor.

B – Alternativa correta.

JUSTIFICATIVA. O diagrama de classes cumpre todos os requisitos levantados a partir da tabela do enunciado.

C e D – Alternativas incorretas.

JUSTIFICATIVA. Os diagramas exigem pelo menos um contato para cada responsável, destacado pela multiplicidade 1..* na extremidade da classe Contato.

E – Alternativa incorreta.

JUSTIFICATIVA. Nesse diagrama, cada aluno se matricula para cursos, e não para turmas. Nesse caso, existe a possibilidade de um aluno se matricular para um curso que não tem turmas abertas. Além disso, cada professor é associado a horários, e não diretamente a turmas. Essa disposição faz com que cada horário possa ser preenchido por apenas um professor.

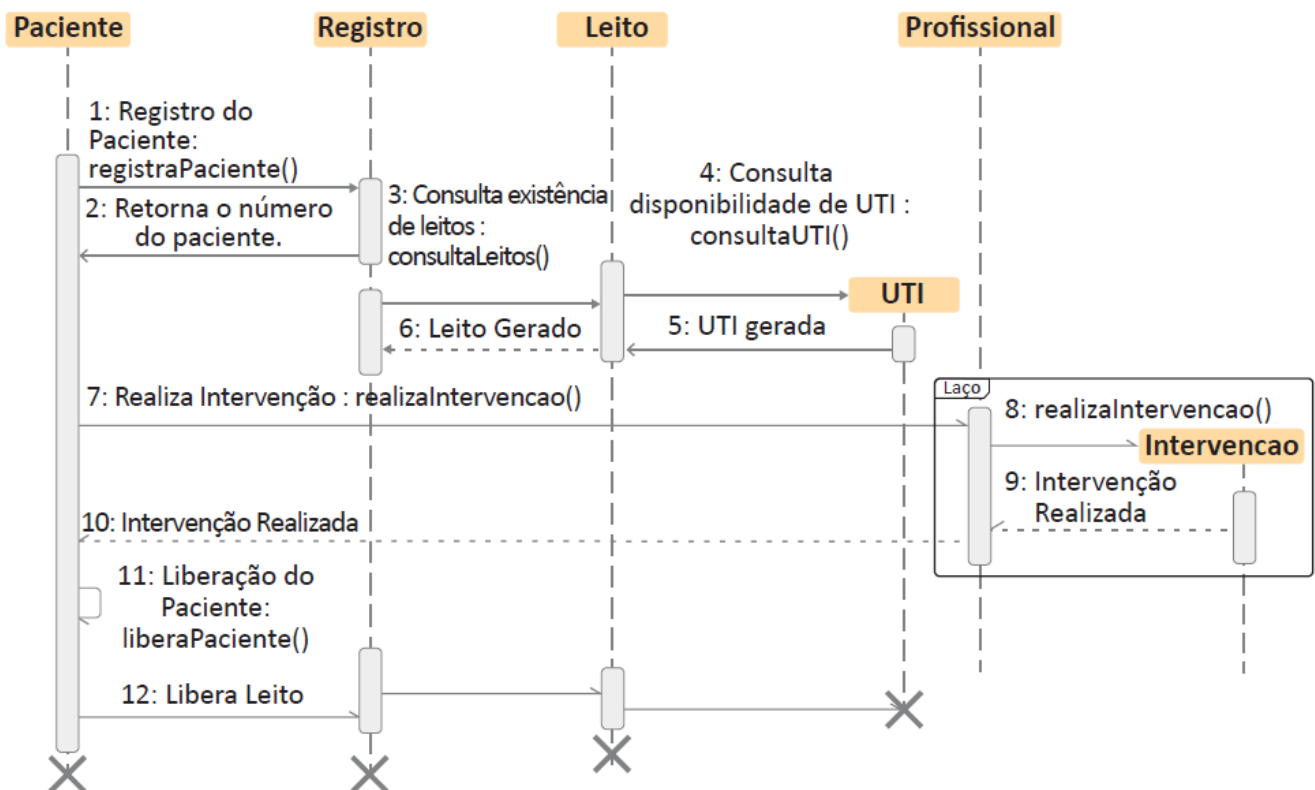
4. Indicações bibliográficas

- FOWLER, M. *UML essencial: um breve guia para a linguagem-padrão de modelagem de objetos*. Porto Alegre: Bookman, 2005.
- LEE, R. C.; TEPFENHART, W. M. *UML e C++: guia prático de desenvolvimento orientado a objeto*. São Paulo: Makron Books, 2001.
- SPACE PROGRAMMER. *Introdução às classes, associações e generalizações*. Disponível em <https://spaceprogrammer.com/uml/introducao-as-classes-associacoes-e-generalizacoes/>. Acesso em 19 abr. 2025.

Questão 9

Questão 9.⁹

Uma ONG decidiu construir um hospital de campanha para tratamento de pacientes diagnosticados com covid-19. Para auxiliar na gerência hospitalar, a ONG contratou alguns programadores voluntários para desenvolver um Sistema de Suporte à Decisão para Gestão Hospitalar. Esse sistema irá auxiliar no registro de todos os procedimentos diários realizados no paciente que dá entrada no hospital, desde sua internação até a saída, seja essa saída por recebimento de alta, por transferência ou por óbito. O sistema foi todo desenvolvido em Java de acordo com o Paradigma Orientado a Objetos. Durante o curto processo de análise, devido à urgência, foram construídos diversos diagramas em UML (Unified Modeling Language). Um desses diagramas relaciona a interação entre os objetos do sistema, o Diagrama de Sequência. Esse diagrama é apresentado a seguir.



Considerando o Diagrama de Sequência apresentado, assinale a opção correta.

- A. Leito é uma subclasse de UTI.
- B. O método registraPaciente() é implementado por Paciente.
- C. Todos os objetos foram criados no instante de execução do Caso de Uso representado.
- D. O diagrama apresenta um erro ao não representar as mensagens de retorno depois da destruição dos objetos.

⁹Questão 16 – Enade 2021 (com adaptações).

- E. A mensagem 6 pode ser substituída pelo estereótipo <<create>> sem causar prejuízo à interpretação correta do Diagrama de Sequência representado.

1. Análise da questão

A questão cobra conhecimentos a respeito de diagramas de sequência, um dos tipos de diagramas UML (Unified Modeling Language). A seguir, faremos uma breve introdução ao tema e, em sequência, analisaremos o diagrama do enunciado.

1.1. Diagrama de sequência

No contexto dos diagramas UML, os **diagramas de sequência** são aqueles capazes de modelar a sequência de mensagens trocadas entre os objetos de um programa computacional, ao longo de uma interação.

Os principais elementos dispostos em um diagrama de sequência são elencados a seguir, com suas respectivas definições.

- Linhas de vida: representam objetos que participam de uma interação. Desse modo, cada instância é representada por uma linha de vida. Graficamente, são desenhadas como linhas verticais.
- Mensagens: são elementos que definem um tipo distinto de comunicação entre instâncias, em uma interação. Cada mensagem carrega informações de uma instância a outra. No diagrama, as mensagens são textos numerados que ocorrem entre as linhas de vida, sendo que as setas que conectam as linhas de vida indicam a ordem das mensagens transmitidas.
- Barras de ativação: indicam o período em que uma instância está ativa e processando uma mensagem. Essas barras são vistas no diagrama sobre as linhas de vida.
- Fragmentos combinados: são agrupamentos lógicos que contêm estruturas condicionais. No diagrama, são destacados por retângulos.

1.2. Análise do diagrama da questão

Com base no diagrama de sequência fornecido no enunciado da questão, a interação entre as instâncias do sistema pode ser resumida conforme os itens a seguir.

1. A instância *Paciente* inicia a interação invocando o método `registraPaciente()`. Isso representa um paciente sendo registrado no sistema, quando dá entrada no hospital.
2. A instância *Registro* retorna o número do paciente, que pode ser usado para identificar o paciente no sistema.
3. A instância *Registro* consulta a existência de leitos disponíveis no hospital por meio do método `consultaLeitos()`.
4. A instância *Registro* também consulta a disponibilidade de UTI pelo método `consultaUTI()`. Isso pode ser necessário, caso o paciente precise de cuidados intensivos.
5. Como resultado das consultas, as instâncias *Leito* e *UTI* são geradas.
6. A instância *Profissional* realiza uma intervenção no paciente por meio do método `realizaIntervencao()`. Isso caracteriza algum tipo de procedimento médico que é realizado no paciente.
7. Depois de sair do laço de intervenção, o paciente é liberado pelo método `liberaPaciente()`. Isso pode representar o paciente recebendo alta, por exemplo.
8. Por fim, o leito que havia sido alocado ao paciente é liberado. Isso indica que o leito ficou disponível para outro paciente.

2. Análise das alternativas

A – Afirmativa incorreta.

JUSTIFICATIVA. A hierarquia entre classes não é exposta no diagrama de sequências. Desse modo, não é possível afirmar se uma classe do sistema é subclasse de outra.

B – Afirmativa incorreta.

JUSTIFICATIVA. Pelo diagrama, sabemos que a instância *Paciente* invoca o método `registraPaciente()`. No entanto, isso não significa, necessariamente, que o método é implementado por *Paciente*. Pode ser que esse método esteja implementado em outra classe, e é apenas chamado por *Paciente*.

C – Afirmativa incorreta.

JUSTIFICATIVA. O diagrama nos mostra que os objetos *Leito* e *UTI* são gerados durante a execução do caso de uso, conforme comentamos na análise do diagrama de sequência. No

entanto, não há indicação de que as instâncias Paciente, Registro e Profissional são criadas durante a execução do caso de uso.

D – Afirmativa incorreta.

JUSTIFICATIVA. A destruição dos objetos, representada no diagrama por símbolos do tipo **x**, indica o momento em que a memória alocada para a criação das instâncias é liberada. Em um diagrama de sequência, não é obrigatório mostrar mensagens de retorno após a destruição de um objeto. Portanto, isso não é considerado um erro.

E – Afirmativa correta.

JUSTIFICATIVA. Alguns diagramas de sequência utilizam estereótipos. O estereótipo <<create>> indica a criação de um objeto. No diagrama de sequência do enunciado, a mensagem 6 diz "Leito Gerado", o que indica a criação do objeto Leito. Portanto, ela poderia ser substituída pelo estereótipo <<create>> sem causar prejuízo à interpretação correta do diagrama.

3. Indicações bibliográficas

- FOWLER, M. *UML essencial: um breve guia para a linguagem-padrão de modelagem de objetos*. Porto Alegre: Bookman, 2005.
- IBM. *Diagramas de sequência*. Disponível em <<https://www.ibm.com/docs/pt-br/rsm/7.5.0?topic=uml-sequence-diagrams>>. Acesso em 31 mai. 2025.
- LEE, R. C.; TEPFENHART, W. M. *UML e C++: guia prático de desenvolvimento orientado a objeto*. São Paulo: Makron Books, 2001.

ÍNDICE REMISSIVO

Questão 1	Legislação profissional. Lei Geral de Proteção de Dados Pessoais (LGPD). Fundamentos da disciplina da proteção de dados pessoais.
Questão 2	Interface Humano-Computador. Responsividade. Design participativo.
Questão 3	Redes de computadores. Endereço IP. Máscara de sub-rede.
Questão 4	Empreendedorismo. Modelo de negócio. Ferramenta Canvas.
Questão 5	Engenharia de software. Processo de desenvolvimento de software. Níveis de maturidade do CMMi.
Questão 6	Banco de dados. Linguagem SQL. Cláusulas SQL.
Questão 7	Modelagem de software. Diagramas UML. Diagramas de caso de uso.
Questão 8	Modelagem de software. Diagramas UML. Diagramas de classes.
Questão 9	Modelagem de software. Diagramas UML. Diagramas de sequência.