Collaborate

Notas

Semana 1

Semana 2

Semana 3

Semana 4

Semana 5

Semana 6

Semana 7

Semana 8

Orientações para

Orientações para

Documentos e

Gabaritos

informações gerais

Referências da disciplina

Facilitadores da disciplina

Repositório de REA's

realização da prova

realização do exame

Calendário Lives

Menu das Semanas

```
Revisar envio do teste: Semana 3 - Atividade Avaliativa
Data de vencimento
                   25/08/23 05:00
                    Completada
Status
Resultado da tentativa 10 em 10 pontos
                    2 horas, 8 minutos
Tempo decorrido
Instruções
                    Olá, estudante!
                       1. Para responder a esta atividade, selecione a(s) alternativa(s) que você considerar correta(s);
                       2. Após selecionar a resposta correta em todas as questões, vá até o fim da página e pressione "Enviar teste".
                       3. A cada tentativa, você receberá um conjunto diferente de questões.
                    Pronto! Sua atividade já está registrada no AVA.
                   Todas as respostas, Respostas enviadas, Respostas corretas, Comentários, Perguntas respondidas incorretamente
Resultados exibidos
   Pergunta 1
                                                                                                                                               1,43 em 1,43 pontos
             Os processos de software são complexos e, como processos intelectuais e criativos, dependem da tomada de decisão e do
             julgamento das pessoas. Um modelo de processo ou modelo de ciclo de vida é um modelo descritivo que fornece as melhores
             práticas para realizar o desenvolvimento de software (ou seja, para realizar o processo de software).
             Considerando o apresentado, avalie as asserções a seguir e a relação proposta entre elas.
             I. O desenvolvimento profissional de software necessita ser gerenciado, baseado num planejamento que deve ser aplicado a todos os
             processos.
                                                                                     POIS
             II. Ainda que não exista um padrão universal de processo de software, sempre há espaço para que se melhorem os processos
             presentes nas muitas organizações.
             A respeito dessas asserções, assinale a alternativa correta.
              Resposta Selecionada: 👩 c. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa para a I.
              Respostas:
                                      a. As asserções l e ll são proposições verdadeiras, e a ll é uma justificativa para a l.
                                      b. A asserção I é uma proposição verdadeira, enquanto a II é falsa.
                                   <sub>C.</sub> As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa para a I.
                                      d. A asserção II é uma proposição verdadeira, enquanto a I é falsa.
                                      e. As asserções I e II são proposições falsas.
              Comentário da
                             JUSTIFICATIVA
              resposta:
                             A asserção I é verdadeira, pois todo e qualquer projeto ou construção de software precisam ser gerenciados em todas as
                             etapas, sempre frisando que testes devem ser realizados não apenas no final mas também em todo o processo, de forma a
                             diminuir retrabalho e custos conforme possibilidades. A asserção II é verdadeira e não justifica a II, pois essa evolução
                             geralmente nomeamos de melhoria contínua em todas as áreas de projetos, mas não em software.
   Pergunta 2
                                                                                                                                               1,43 em 1,43 pontos
             Conforme sentencia o PMBOK (2017), "um projeto é um esforço temporário que tem como finalidade um resultado único e possui recursos delimitados".
             PMI. Um guia do conhecimento em gerenciamento de projetos. Guia PMBOK, 6. ed. EUA: Project Management Institute, 2017.
             Levando em consideração as características inerentes a um projeto de qualquer natureza, identifique se são (V) verdadeiras ou (F) falsas as afirmativas a seguir.
             I. ( ) Deve ter data de início, pausa e fim definidas.
             II. ( ) Tem a intenção de criar um novo produto ou serviço, ou ainda aprimorar algo já existente.
             III. ( ) Precisa ter um escopo inicial bem detalhado, para que as datas de início e fim sejam cumpridas sem alterações.
             IV. ( ) Deve ter estimativas dos custos e recursos que serão utilizados, como profissionais, materiais e infraestrutura.
             Assinale a alternativa que apresenta a sequência CORRETA.
              Resposta Selecionada: e. F - V - F - V.
                                        F - V - V - V.
              Respostas:
                                      a.
                                      b. F-F-F-V.
                                      c. V - V - V - V.
                                      d. V - F - V - F.
                                   🌠 e. F - V - F - V.
              Comentário da JUSTIFICATIVA
              resposta:
                             Um projeto é, obrigatoriamente, temporário, pois deve ter datas de início e fim definidas, mas não precisa ter datas de pausa estabelecidas. A
                             intenção de um projeto pode ser criar algo novo, ou aprimorar um produto ou serviço já existente. É imprescindível que exista ao menos um
                             escopo mínimo de projeto, porém as datas de início e fim podem sofrer alterações – e faz parte do gerenciamento de projetos administrar tais
                             mudanças. Por fim, é necessário estipular custos e recursos ao iniciar o projeto e realizar seu acompanhamento durante a execução.
   Pergunta 3
                                                                                                                                               1,44 em 1,44 pontos
             Os processos de desenvolvimento de "software dirigidos por planos que especificam completamente os requisitos e depois projetam,
             constroem e testam um sistema não são voltados para o desenvolvimento rápido de software. À medida que os requisitos mudam ou
```

```
que problemas de requisitos são descobertos, o projeto ou a implementação do sistema precisam ser retrabalhados e testados
novamente. Como consequência, um processo convencional em cascata ou baseado em especificação normalmente é demorado e o
software final é entregue ao cliente muito depois do prazo originalmente estipulado" (SOMMERVILLE, 2019, p. 58).
SOMMERVILLE, I. Engenharia de software. 10. ed. São Paulo: Pearson, 2019.
Baseando-se no conceito de desenvolvimento ágil, analise as asserções a seguir e a relação proposta entre elas.
I. O sistema é desenvolvido numa série de entregas em que usuários finais e clientes em geral são incluídos em desenvolvimento e
avaliação de cada uma.
                                                             POIS
```

II. As abordagens ágeis de desenvolvimento de software apontam projeto, comunicação e implementação como as principais

A asserção I é verdadeira, pois satisfazer o cliente por meio de entrega antecipada e contínua de software é uma ação

primordial. Precisamos estar aptos a mudanças para a vantagem competitiva de clientes, mesmo no final do

A asserção II é uma proposição verdadeira, enquanto a I é falsa. 👩 d. As asserções l e ll são proposições verdadeiras, mas a ll não é uma justificativa para a l. e. A asserção I é uma proposição verdadeira, enquanto a II é falsa.

b. As asserções I e II são proposições verdadeiras, e a II é uma justificativa para a I.

Resposta Selecionada: 👩 d. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa para a I.

a. As asserções I e II são proposições falsas.

```
desenvolvimento. A asserção II é verdadeira, pois é necessária a colaboração diária entre empresários e desenvolvedores
ao longo do projeto. Construir projetos em torno de indivíduos motivados é outro ponto importante, criando ambiente e
suporte de que os desenvolvedores precisam (para que possuam confiança para fazer o trabalho). Todavia a asserção II não
é uma justificativa para a I.
```

atividades no processo de software.

JUSTIFICATIVA

Respostas:

Comentário

da resposta:

(4) Refatoração.

(5) Propriedade coletiva.

devem ter sucesso.

Respostas:

Comentário

da resposta:

Pergunta 5

responsabilidade por todo código.

frequência que acrescente funcionalidades à primeira versão de forma incremental.

A sequência correta, de cima para baixo, da associação entre item e descrição é:

ob. 1-II; 2-IV; 3-V; 4-III; 5-I.

c. 1-V; 2-IV; 3-II; 4-I; 5-III.

d. 1-I; 2-IV; 3-III; 4-V; 5-II.

e. 1-II; 2-I; 3-IV; 4-III; 5-V.

computador e na facilidade de uso do software.

profissionais desenvolverem um *software* de altíssima qualidade.

Fonte: PRESSMAN; MAXIM (2016), p.16.

PORQUE

Respostas:

Comentário da

Respostas:

Pergunta 7

afirmativas a seguir.

iterações.

Respostas:

enfoque em subprodutos distintos.

centrados no fluxo ou no incremento de funcionalidades.

Assinale a alternativa que apresenta a sequência **CORRETA**.

V - V - F - V - V.

V - F - V - F - F.

c. F - F - F - V.

F-V-V-F.

a.

🕜 e.

JUSTIFICATIVA

a.

b.

() C.

JUSTIFICATIVA

A respeito dessas asserções, assinale a alternativa correta.

Pergunta 4 1,44 em 1,44 pontos Leia o trecho a seguir, retirado da obra de SOMMERVILLE, chamada **Engenharia de Software**: "Na prática, a aplicação da Programação Extrema como proposta originalmente se provou mais difícil do que o previsto. Na XP os requisitos são expressos em cenários (chamados de histórias do usuário) implementados diretamente como uma série de tarefas. Os programadores trabalham em pares e desenvolvem testes para cada tarefa antes de escreverem o código. Todos os testes devem ser executados com sucesso quando o novo código é integrado ao sistema, já que há um curto intervalo de tempo entre os lançamentos (*releases*) do sistema." (SOMMERVILLE, 2019, p. 62) Fonte: SOMMERVILLE, I. **Engenharia de software**. 10. ed. São Paulo: Pearson, 2019. p. 62.

```
Com base na prática chamada de programação extrema, associe os conceitos a seguir com as respectivas descrições mais adiante.
(1) Integração contínua.
(2) Programação em pares.
(3) Lançamentos pequenos.
```

III. Esse processo deve ser feito por todos os desenvolvedores no código de forma contínua, logo sejam encontradas possíveis melhorias para ele. Isso mantém o código simples e de fácil manutenção. IV. Forma de trabalho da equipe de desenvolvimento cujo objetivo é realizar a conferência de trabalhos de modo geral entre os próprios membros da equipe, oferecendo apoio para um resultado sempre satisfatório. V. Relacionado à maneira por meio da qual sistemas são disponíveis, a fim de agregar valor ao negócio, desenvolvido em primeiro lugar. É necessária uma

I. Os pares de desenvolvedores trabalham em todas as áreas do sistema, de modo que não se desenvolvam "ilhas de conhecimento" e todos assumam a

II. Assim que o trabalho numa tarefa é completo, é integrado ao sistema completo. Após qualquer integração desse tipo, todos os testes de unidade do sistema

Resposta Selecionada: _{b.} 1-II; 2-IV; 3-V; 4-III; 5-I. a. 1-IV; 2-II; 3- I; 4-V; 5-III.

A sentença I se encaixa no conceito 5, pois a **propriedade coletiva** do código aponta ao fato de que, em processo ágil de

pequenos lançamentos (releases) são essenciais na XP: mostram o rigor da equipe em relação ao projeto, visto que, no

Segundo Maxim e Pressman (2016), a engenharia de software abrange um processo, um conjunto de práticas e um leque de ferramentas que possibilitam aos

Processo

Foco na qualidade

momento do estágio de planejamento, havia visualizações/ideias estimadas da reação do cliente com base no rendimento do

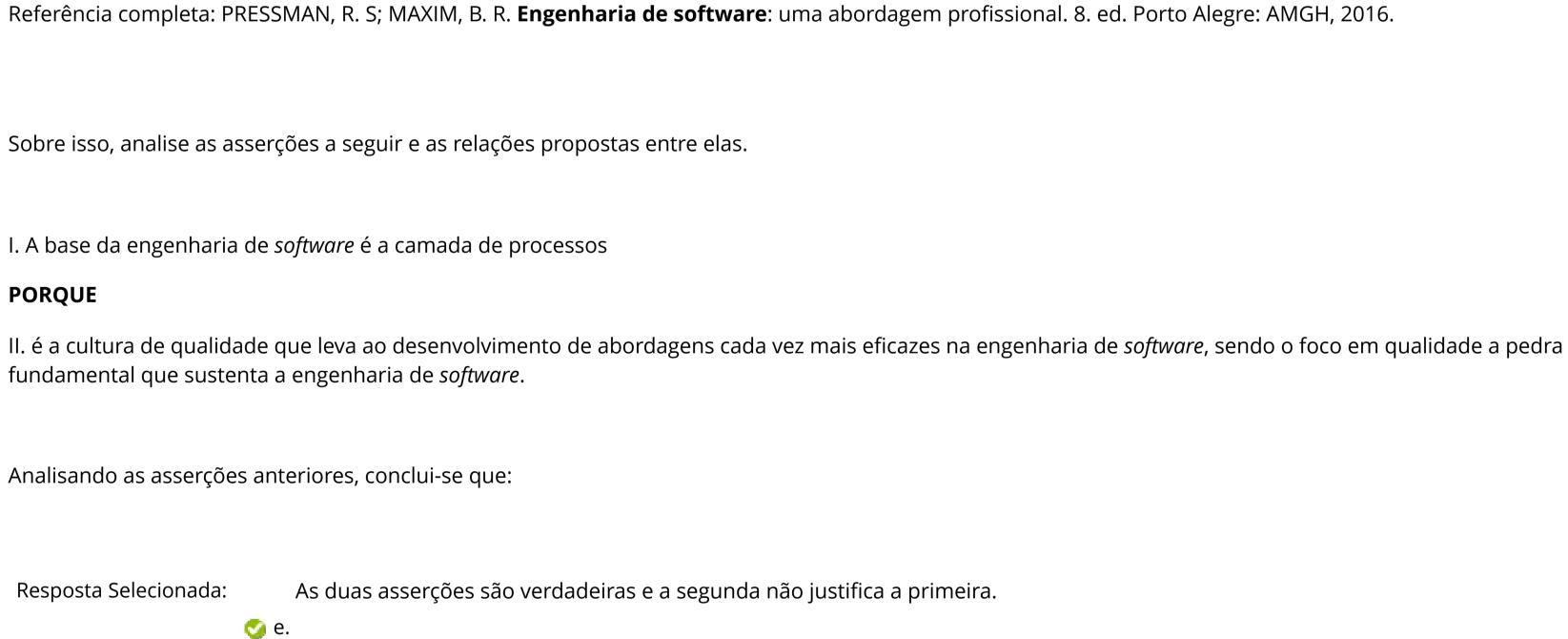
1,42 em 1,42 pontos

1,42 em 1,42 pontos

1,42 em 1,42 pontos

```
desenvolvimento de software, nenhum membro da equipe possui parte específica do código-fonte. Portanto qualquer membro
da equipe pode contribuir com nova ideia ao projeto para melhorar o design do software e corrigir quaisquer erros no projeto. A
sentença II se encaixa no conceito 1, pois, na programação extrema (XP), a integração contínua evita ou detecta
problemas de compatibilidade antecipadamente; a integração é um tipo de atividade "pague-me agora ou pague-me mais
tarde". Se você integrar ao longo do projeto em pequenas quantidades, não se encontrará tentando integrar no sistema por
semanas ao final do projeto, enquanto o prazo passa. A sentença III se encaixa no conceito 4, pois a refatoração tem como
principal objetivo tornar o código inicial mais simples. Normalmente, aliás, o código torna-se cada vez mais complexo durante
o processo de desenvolvimento de software. A sentença IV se encaixa no conceito 2, pois a programação em pares é uma
habilidade social que leva tempo para ser aprendida. Você está buscando uma forma cooperativa de trabalhar que inclua dar
e receber de ambos os parceiros, independentemente do status corporativo. A sentença V está para o conceito 3, pois
```

PRESSMAN, R. S; MAXIM, B. R. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: AMGH, 2016. Portanto, a engenharia de *software* é definida em quatro camadas distintas, conforme ilustra a figura abaixo: **Ferramentas** Métodos



Entretanto, o foco em qualidade não é explicação ou justificativa para os processos serem a base de execução, pois tal justificativa significa

As duas asserções são verdadeiras e a segunda justifica a primeira.

As duas asserções são verdadeiras e a segunda não justifica a primeira.

b. A primeira asserção é verdadeira e a segunda é falsa.

A primeira asserção é falsa e a segunda é verdadeira.

```
resposta:
                 O processo de engenharia de software é a liga que mantém as camadas de ferramentas e métodos coesas e possibilita o desenvolvimento
                 de software de forma racional e dentro do prazo. Por isso, a primeira asserção é verdadeira.
                 Todas as abordagens de engenharia devem fundamentar-se em um comprometimento organizacional com a qualidade, não sendo
                 diferente para a engenharia de software. Portanto, a segunda asserção é verdadeira, uma vez que a pedra fundamental que sustenta as
```

outras três camadas é o foco na qualidade de software.

conectar as camadas de métodos e ferramentas.

C. As duas asserções são falsas.

```
Pergunta 6
          Projetos de desenvolvimento de software têm especificidades inerentes à área da programação e outras tecnologias envolvidas, assim como projetos de
       engenharia para a construção de prédios ou projetos para a criação de um novo produto cosmético também apresentam suas características particulares
          inerentes ao produto ou serviço a ser desenvolvido.
          Considere que um novo software precisa ser desenvolvido para a gestão de armazéns de grãos de soja e milho. Assinale a alternativa que apresenta o melhor
          conjunto de características e habilidades para contratar o gerente desse projeto:
           Resposta
                                 ② C.
                                 Graduado em análise e desenvolvimento de sistemas, capacidade de liderança, eficiente na comunicação e com visão de longo prazo.
           Selecionada:
```

comunicação, visão de longo prazo, persistência. Logo,

```
Graduado em análise e desenvolvimento de sistemas, capacidade de liderança, eficiente na comunicação e com visão de curto prazo.
                JUSTIFICATIVA
Comentário da
resposta:
                 Normalmente, um gerente de projetos tem formação acadêmica nas áreas da computação, embora existam diversos profissionais bem
                 capacitados com formação em outras áreas. Portanto, para esse contexto, ser graduado tanto em análise e desenvolvimento de sistema
                quanto em engenharia agronômica são requisitos igualmente válidos.
```

Graduado em engenharia agronômica, volúvel, eficiente na comunicação e com visão de longo prazo.

d. Graduado em engenharia agronômica, capacidade de liderança, persistente e com visão de curto prazo.

Já as soft skills desejadas para um gerente de projeto de software são: capacidade de liderança e de desenvolver equipes, eficiência na

Graduado em engenharia agronômica, capacidade de desenvolver equipes, eficiente na comunicação e com visão de curto prazo.

Graduado em análise e desenvolvimento de sistemas, capacidade de liderança, eficiente na comunicação e com visão de longo prazo.

```
testes.
   Para o desenvolvimento de software, é aconselhável tratá-lo como projeto, no qual, comumente, as etapas são divididas em: início, organização, execução e fim.
   SOMMERVILLE, I. Engenharia de Software. 9. ed. São Paulo: Pearson, 2011.
```

Segundo Sommerville (2011), o processo de *software* é entendido como um conjunto de atividades responsáveis pela especificação, projeto, implementação e

Levando em consideração os distintos tipos de ciclos de vida de um projeto de desenvolvimento de software, identifique se são (V) verdadeiras ou (F) falsas as

I. () Os ciclos de vida **preditivos** determinam claramente o escopo, prazo e custo do projeto. Normalmente, são organizados em fases sequenciais, com

```
II. ( ) Os ciclos de vida iterativos repetem atividades do projeto em fases, e em cada uma delas a equipe do projeto aumenta seu entendimento do produto.
Essas iterações tendem a melhorar os entregáveis, ou ainda criar novos, em que o produto final será a acumulação de funcionalidades construída nessas
III. ( ) Os ciclos de vida preditivos servem para gerenciar objetivos pouco definidos ou de alta complexidade, ou ainda quando a entrega parcial do produto é
algo fundamental para a consolidação das ideias e o consenso de sucesso do projeto.
```

```
V. ( ) Os ciclos de vida adaptativos são mais adequados que os ciclos preditivos para projetos em que o escopo do produto ou serviço a ser entregue está bem
definido e existe pleno conhecimento de como se deve desenvolver a solução.
```

IV. () Os ciclos de vida **adaptativos** são similares aos **iterativos**, porém com delimitações muito claras das iterações e seus objetivos – que podem ser

```
Resposta Selecionada:
                             V - V - F - V - F.
                        🧭 e.
```

```
d.
                           V - V - F - V - F.
                      🕜 e.
              JUSTIFICATIVA
Comentário
da resposta:
              Ciclos de vida preditivos têm escopo, prazo e custo definidos de forma rígida, por isso, são recomendados para projetos em que o escopo do
              produto ou serviço a ser entregue está bem definido. Já os ciclos iterativos são recomendados para projetos com objetivos pouco definidos,
              ou com alta complexibilidade e pouco conhecimento de como construir a solução, pois sua natureza incremental permite à equipe se
              retroalimentar de conhecimentos e incrementar a experiência durante a execução do projeto. Por fim, os ciclos de vida adaptativos, também
              conhecidos como métodos ágeis, são indicados para a participação contínua e ordenada dos interessados do projeto, com iterações bem
              estruturadas e que tem por objetivo, ao final de cada iteração, garantir que o produto esteja pronto para a revisão do cliente.
```