



Critérios de Correção

MODELAÇÃO DE SISTEMAS DE INFORMAÇÃO | CÓDIGO 21177

Período de Realização: decorre 23-06-2020 deste 10:00 com 3 horas de duração

Data de Limite de Entrega: decorre 23-06-2020 até 13:00 de Portugal Continental

Temática / Tema / Conteúdos: Modelação de Sistemas de Informação

Objetivos: Modelizar requisitos de sistemas de informação com UML "Unified Modeling Language" usando diagramas de utilização, diagramas de classes e CRUD. Especificar o software com diagramas de interação: sequência, estados e atividades; e especificar infraestruturas com os diagramas de arquitetura.

Trabalho a desenvolver: Resolução de um conjunto de exercícios.

Critérios de avaliação e cotação: A cotação deste e-fólio é de 120 pontos = 12 valores, pode encontrar as cotações parciais junto de cada pergunta. A interpretação das perguntas também faz parte da sua resolução, se encontrar alguma ambiguidade deve indicar claramente como foi resolvida. Critérios de avaliação gerais: (i) para a dificuldade de leitura (linhas cruzadas, letras com fontes desadequadas) a penalização é de 20% a 100%; (ii) para erros e omissões a penalização é de 20% a 100%.

Normas a respeitar: Deve redigir o seu E-fólio na Folha de Resolução disponibilizada na turma e preencher todos os dados do cabeçalho. Podem ser incluídas imagens e digitalizações de conteúdos produzido manualmente pelo estudante. Todas as páginas do documento devem ser numeradas. O seu E-fólio não deve ultrapassar 1 página por pergunta. O documento A4 deve ser redigido em Times New Roman, tamanho de letra 12. O espaçamento entre linhas deve corresponder a 1,0 ou 1,5 linhas. Nomeie o ficheiro com o seu número de estudante, seguido da identificação do E-fólio, segundo o exemplo apresentado: 00000efolioGlobal. Finalmente deve gerar um PDF do documento. Deve carregar o referido ficheiro para a plataforma no dispositivo E-fólio Global até à data e hora limite de entrega. Evite a entrega próximo da hora limite para se precaver contra eventuais problemas. O ficheiro a enviar não deve exceder 8 MB. Votos de bom trabalho! Luís Cavique.

Vetor das perguntas: 1 2 3, 4 5 6

Vetor das cotações: 2 2 2, 2 2 2 somando 12 valores

Parte I – Caso Prático Integrado

Leia com atenção a seguinte introdução. Na sua análise não exceda 5 atores, 7 casos-de-utilização e 7 classes.

Considere um subsistema de gamificação de um 'contact-center'. No 'contact-center' os funcionários tratam as solicitações dos clientes através de mensagens de voz e de texto. O subsistema de gamificação recebe informação do subsistema de mensagens. O subsistema de gamificação tem os seguintes requisitos:

- Os utilizadores têm um identificador, nome e contactos;
- Os utilizadores agrupam-se em equipas; cada equipa tem um identificador, um nome e um chefe de equipa;
- Existem ainda atividades semanais de gamificação como os desafios ('challenges') e as missões ('quests');
- Os desafios têm um nome, número da semana (1 a 52), ano e um KPI ('key performance indicator'); um exemplo de desafio será responder a uma determinada quantidade de solicitações de clientes numa determinada semana; os desafios estão associados aos utilizadores;
- Por outro lado, as missões estão associadas às equipas e devem ser registados os atributos: nome, semana, ano e KPI da equipa;
- Os utilizadores que atingem os KPI são distinguidos com distintivos ('badges') e as equipas são recompensadas com taças ('cup');

Quem regista as missões das equipas é o diretor. Quem regista os desafios individuais é o chefe da equipa. Todos os utilizadores podem visualizar os seus resultados-semanais individuais e da sua equipa. Os KPI são atualizados pelo subsistema de mensagens. Todas as semanas o diretor elege o empregado/utilizador da semana e a equipa da semana com base nos distintivos e nas taças. Pretende-se que o subsistema de gamificação tenha um 'leaderboard' para os utilizadores que acumula a informação semanal.



1. (2,0 valores) O que entende por requisitos funcionais e não-funcionais. Defina exemplos de requisitos funcionais e não-funcionais do caso em estudo.

Resposta:

Requisitos funcionais: são os requisitos associados às funcionalidades (use-cases) do sistema, i.e., às funcionalidades que o sistema deve suportar.

Requisitos não-funcionais: são requisitos que não são suportados pelo sistema, sendo transversais ou complementares ao disto sistema tais como requisitos organizacionais (políticas, processos), externos (ético, legais) ou de produto (eficiência, portabilidade).

Exemplos de requisitos funcionais:

- registar as missões semanais
- atualizar os KPI

Exemplos de requisitos não-funcionais:

- políticas de acesso e segurança dos dados
- portabilidade da aplicação de gamificação

Crítérios de correção:

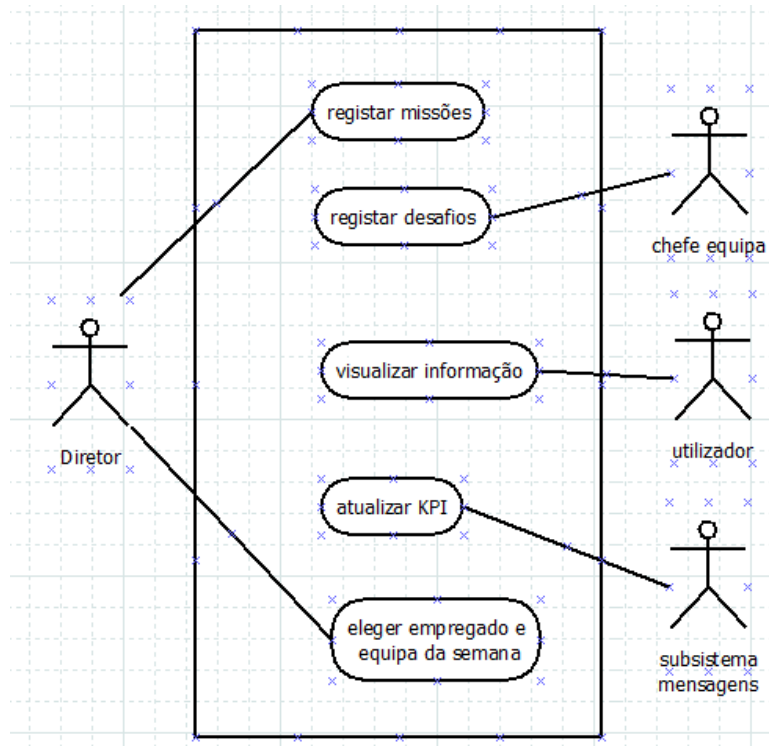
- 50% para a definição e exemplos de requisitos funcionais
- 50% para a definição e exemplos de requisitos não-funcionais
- penalização de 50% a 100% para requisitos desadequados

2. (2,0 valores) Construa a Casos-de-Utilização versus Atores (ver tabela em baixo) e o Diagrama de Casos-de-Utilização do novo sistema.

Casos-de-uso \ Atores	A1	A2	A3
X			
Y			
Z			
W			

Resposta:

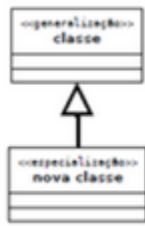
casos-de-uso \ atores	diretor	chefe equipa	utilizador	subsistema mensagens
registar missões	X			
registar desafios		X		
visualizar informação			X	
atualizar KPI				X
eleger empregado/equipa	X			



Crítérios de correção:

- 50% para a matriz
- 50% para o diagrama
- penalização de 50% a 100% para atores ou casos de uso desadequados

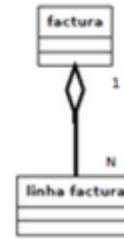
3. (2,0 valores) Construa o Diagrama de Classes do novo sistema com os seguintes elementos:



Herança
Generalização-Especialização

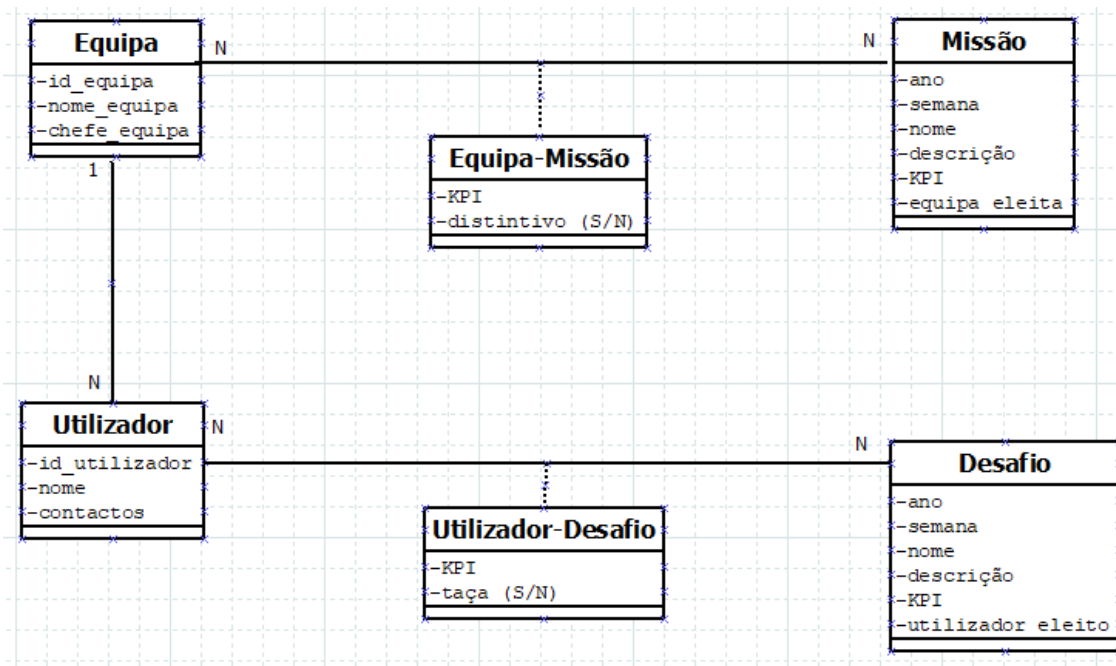


Classe
Associativa



Agregação
Todo-Parte

Resposta:



Critério de correção:

- 70% para as classes
- 30% para as associações
- penalização de 50% a 100% para classes ou associações desadequados

4. (2,0 valores) Construa a matriz CRUD (*create, read, update, delete*), com os Casos-de-Utilização versus as Classes encontradas, com o seguinte aspeto:

Casos-de-uso \ Classes	A	B	C	D
X				
Y				
Z				
W				
Contadores C R U D				

Resposta:

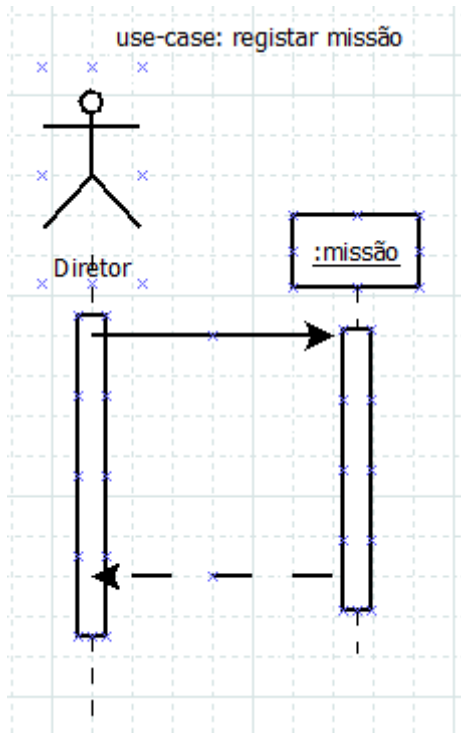
casos-de-uso \ classes	equipa	utilizador	missão	desafio	missão semanal	desafio semanal
registar missões			CRUD			
registar desafios				CRUD		
visualizar informação	R	R	R	R	R	R
atualizar KPI					CRUD	CRUD
eleger empregado/equipa semana			RU	RU		
outros	CRUD	CRUD				
contadores CRUD	1211	1211	1321 (a)	1321 (a)	1211	1211
(a) update realizado sobre atributos diferentes						

Critério de correção:

- 70% para a tabela CRUD
- 30% para os contadores CRUD
- penalização 50% a 100% para casos-de-uso, classes ou preenchimento desadequados

5. (2,0 valores) Construa um Diagrama de Sequência de um caso de utilização do sistema de informação. Tenha em consideração a informação das alíneas anteriores.

Resposta:



Critério de correção:

- 50% classes
- 50% atores e mensagens
- penalização de 50% a 100% para atores, classes ou mensagens desadequadas

Parte II – Outros

6. (2,0 valores) Construa um Diagrama de Atividades que calcule a função “maior subseqüência”. A função recebe um número inteiro N e uma lista contendo números inteiros. Pretendem-se conhecer a maior soma de números consecutivos de dimensão N.

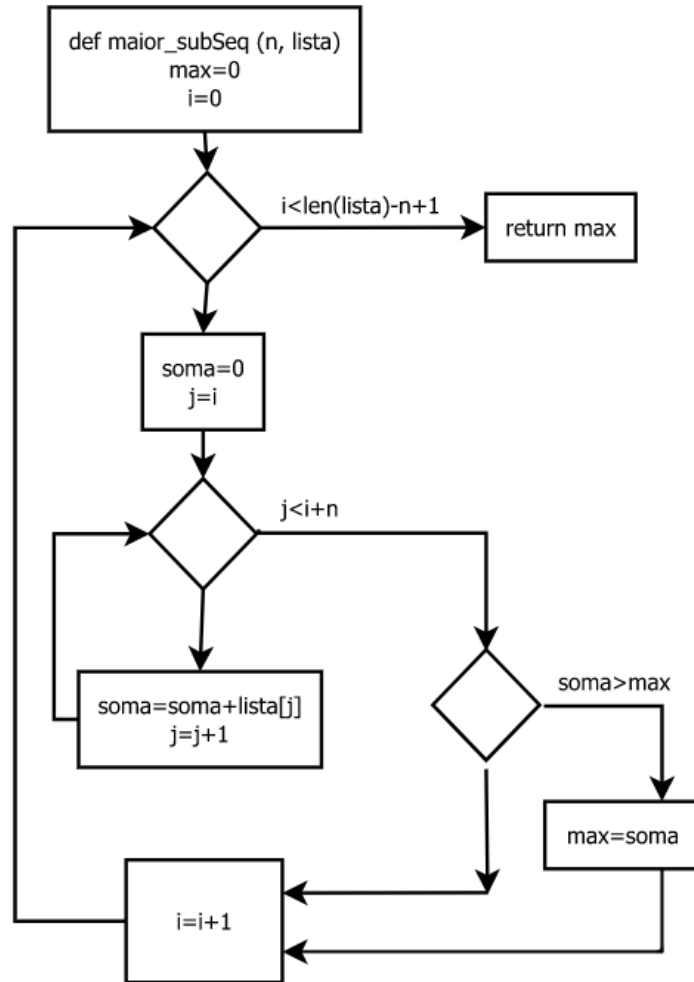
Exemplo 1: maior_subsequência (2, [12,4,2,14,3]) = 17, relativo à subseqüência = [14,3].

Exemplo 2: maior_subsequência (3, [12,4,2,14,3]) = 20, relativo à subseqüência = [4,2,14].

Resposta:

Em Python teremos a função max_subseq com ciclos For e While da seguinte forma:

<pre>def maior_subseq (n,lista): max=0 for i in range(len(lista)-n+1): soma=0 for j in range (i, i+n): soma=soma+lista[j] if soma>max: max=soma return max</pre>	<pre>def maior_subseq2(n,lista): max=0 i=0 while i < (len(lista)-n+1): soma=0 j=i while j < i+n: soma=soma+lista[j] j=j+1 if soma>max: max=soma i=i+1 return max</pre>
---	---



Critério de correção:

- 50% para o ciclo e condições
- 50% para os detalhes do algoritmo: variáveis e afetações

FIM