

NOME:

NUM:

Cada uma das 13 alíneas desta parte do teste contém uma frase inicial que é completada de cinco (5) formas diferentes. Indica, para cada um dos casos, se a frase inicial mais a parte que a completa é verdadeira (v) ou falsa (f). Cada resposta vale 0.15 valores: +0.15 val. se for escolhida a alternativa correta, -0.15 val. se for escolhida a alternativa incorreta. A cotação de cada alínea nunca é negativa, ou seja, no caso do número de respostas incorretas ser igual ou maior que o número de resposta corretas, a alínea é cotada com 0 (zero)

[0.75 val.] **1. Os requisitos dum sistema de engenharia**

- ☒ f correspondem a necessidades dos utilizadores.
- ☒ f incluem as restrições a que esse sistema está sujeito.
- v ☒ têm que ser todos incorporados na solução para o sistema ter alta aceitação no mercado.
- v ☒ não devem ser documentados, caso não sejam incorporados na solução final.
- v ☒ são as capacidades que os utilizadores possuem para satisfazer as suas expectativas.

[0.75 val.] **2. Um requisito funcional**

- v ☒ nunca pode depender de questões tecnológicas.
- v ☒ diz-se implícito se for incluído na documentação pelos engenheiros.
- v ☒ diz-se explícito se foi pedido pelos clientes, mas não está incluído na documentação.
- ☒ f pode ser classificado como requisito não funcional por uma pessoa diferente.
- ☒ f pode ser fortemente modificado devido à incorporação de um novo requisito não funcional.

[0.75 val.] **3. Os requisitos NÃO funcionais dum sistema**

- ☒ f correspondem a restrições que são impostas a esse sistema.
- v ☒ modificam de forma muito significativa as funcionalidades desse sistema.
- ☒ f tendem a ter um impacto em todo esse sistema (e não apenas em alguns dos seus componentes).
- ☒ f são cruciais para definir a sua arquitetura.
- ☒ f incluem as categorias aparência, usabilidade e padronização.

[0.75 val.] **4. Os requisitos NÃO funcionais da categoria**

- v ☒ “disponibilidade” quantificam a fração de tempo que o sistema demora a ser mantido.
- v ☒ “usabilidade” estão relacionados com a facilidade de uso e de manutenção.
- ☒ f “segurança” assumem duas vertentes principais: confidencialidade e integridade.
- ☒ f “legal” implicam envolver juristas e advogados no processo de levantamento.
- ☒ f “cultural” são críticos em produtos comercializados em vários países.

[0.75 val.] **5. A engenharia de requisitos**

- v ☒ é uma disciplina com aplicação exclusiva em engenharia de software.
- ☒ f é uma área multidisciplinar, mas com um âmbito muito restrito.
- ☒ f é um conjunto de atividades que permite levantar, negociar e documentar os requisitos dum sistema em desenvolvimento.
- v ☒ está focada na definição da solução para o problema que a equipa de desenvolvimento enfrenta.
- ☒ f garante que todos os requisitos relevantes serão conhecidos e compreendidos com o necessário nível de detalhe.

[0.75 val.] **6. O levantamento de requisitos**

- ☒ f permite identificar os requisitos a incorporar no sistema.
- v ☒ deve ocorrer apenas no início do projeto, antes das decisões de conceção.
- ☒ f inclui a identificação das partes interessadas.
- ☒ f inclui a priorização dos requisitos funcionais.
- v ☒ não inclui o estudo do domínio do problema, que pertence à fase de “estudo de viabilidade técnica”.

Cotação: 1-13:

14: 1

15:

16:

TOTAL:

[0.75 val.] **7. A escrita de requisitos em português**

- ☒ ☐ f deve recorrer a um vocabulário diversificado, para tirar o melhor partido possível da riqueza dessa língua.
- ☒ ☐ f deve limitar o vocabulário a usar, para permitir o seu processamento automático.
- ☒ ☐ f deve tirar partido dos sinónimos para evitar textos repetitivos.
- ☒ ☐ f deve recorrer ao uso de formatos padronizados, para uniformização.
- ☒ ☐ f não precisa de ter em atenção a ambiguidade.

[0.75 val.] **8. Um modelo de domínio**

- ☒ ☐ f descreve o vocabulário, os conceitos e as características dos sistemas que se podem desenvolver para o domínio considerado.
- ☒ ☐ f descreve a arquitetura genérica dos sistemas dum dado domínio do conhecimento.
- ☒ ☐ f descreve os aspetos comuns dos sistemas que se podem construir num dado domínio.
- ☒ ☐ f descreve o contexto em que um sistema vai operar, assim que for construído.
- ☒ ☐ f identifica e relaciona os conceitos do domínio a que diz respeito.

[0.75 val.] **9. A arquitetura dum sistema de software**

- ☒ ☐ f representa a sua organização fundamental.
- ☒ ☐ f não constitui um aspeto crítico em sistemas de baixa complexidade.
- ☒ ☐ f é fortemente condicionada pelos requisitos funcionais.
- ☒ ☐ f representa a visão macroscópica (i.e., o esqueleto) do sistema.
- ☒ ☐ f estabelece a comunicação entre os seus componentes.

[0.75 val.] **10. Numa abordagem orientada ao risco,**

- ☒ ☐ f o esforço para conceber a arquitetura do sistema deve ser proporcional aos riscos do projeto.
- ☒ ☐ f o esforço para conceber a arquitetura do sistema deve ser menor, se a equipa tiver muito familiarizada com o domínio do projeto.
- ☒ ☐ f o impacto dum risco e a sua probabilidade de ocorrência são, geralmente, inversamente proporcionais.
- ☒ ☐ f o risco é definido como a probabilidade de falha dividida pelo impacto dessa falha.
- ☒ ☐ f a arquitetura tem de ser cuidadosamente analisada, mesmo quando o risco de falha é baixo.

[0.75 val.] **11. Um padrão de conceção (design pattern)**

- ☒ ☐ f permite atacar um problema para o qual existem soluções bem estabelecidas.
- ☒ ☐ f define os atores para a solução de um dado problema.
- ☒ ☐ f é um conjunto de classes reutilizáveis.
- ☒ ☐ f é específico de uma linguagem.
- ☒ ☐ f define os vários compromissos entre possíveis soluções.

[0.75 val.] **12. O padrão Observer**

- ☒ ☐ f delega a funcionalidade variável de visualização para os observadores.
- ☒ ☐ f especifica de forma estática os objetos observadores.
- ☒ ☐ f recorre ao mecanismo de 'push' de informação entre observados e observadores.
- ☒ ☐ f definida apenas uma interface na sua estrutura de classes.
- ☒ ☐ f facilita a extensibilidade de uma solução quando são esperadas alterações e modificação nos observados.

[0.75 val.] **13. Um padrão arquitetural**

- ☒ ☐ f define a arquitetura de um dado padrão de concepção.
- ☒ ☐ f melhora a coerência dos módulos de um sistema.
- ☒ ☐ f define as responsabilidades dos componentes do sistema.
- ☒ ☐ f não é específico de qualquer linguagem de programação.
- ☒ ☐ f tem implicações em toda a estrutura de um sistema de software.

NOME:

NUM:

[3.5 val.] **14.** Considera um sistema para gerir o processo de submissão de pedidos de subsídio escolar (numa câmara municipal). Os pedidos são submetidos por um dos responsáveis pelo agregado (pai, mãe, ou outro) e são revistos pelos fiscais da câmara.

(a) **Reescreve**, nos casos necessários, os seguintes requisitos de utilizador à luz das recomendações dadas nas aulas (sugestão: usa o formato *user stories*).

(b) **Justifica** as alterações que fizeres.

R1: Um dos pais deve submeter, através duma aplicação que se instala no telemóvel e cuja interface gráfica se adapta a esse telemóvel, o seu pedido de subsídio.

R2: O sistema, por regra, atribui, a cada pedido de subsídio submetido, um fiscal, que é escolhido em função da freguesia de residência do agregado familiar.

R3: O número de pedidos que cada fiscal tem ainda por processar, é-lhe enviado, diariamente, através dum email.

R4: O sistema deve permitir a edição de avaliações sobre cada pedido, que são da responsabilidade do respetivo fiscal. O responsável camarário pelo processo de subsídios introduz no sistema a decisão final (aceite/rejeitado/dados insuficientes/fora de âmbito) sobre cada pedido de subsídio com base nas avaliações.

[3.25 val.] **15.** Elabora um diagrama de atividades que modele as tarefas dum pequeno-almoço, que inclui: aquecer chá no fogão, adicionar leite ou açúcar ao chá, tostar pão, colocar fiambre ou queijo no pão tostado, beber chá, comer pão e lavar e secar a louça. Admite que o pão pode ter só queijo, só fiambre, ou os dois. Admite também que o pão pode eventualmente não ser tostado, se estiver fresco. Sempre que possível, as tarefas devem ocorrer em paralelo.

[3.5 val.] **16.** Pretende-se desenvolver um serviço para suportar a negociação de criptomoedas. O serviço deve fornecer funcionalidades de suporte à negociação com as seguintes características:

- Disponibilizar uma lista de criptomodeas que podem ser negociadas por investidores (utilizadores finais).
- Cada uma das criptomoedas tem um valor de negociação que varia com o tempo, este valor é obtido através de um serviço externo.
- Os investidores têm interesse em receber alertas para variações significativas do preço de critptmoedas, podendo escolher alertas para variações superiores a 5% ou 10% ou 20% para intervalos pré-definidos de 10 minutos, uma hora ou quatro horas.
- Cada investidor pode definir quais as criptomoedas para as quais pretende receber os alertas e qual o nível de variação do preço para o qual deseja receber os alertas.

(a) **Conceptualiza** um sistema que permita dar solução a este desafio apresentando um diagrama UML para a solução. Considera o padrão mais adequado para este tipo de problema.

(b) **Especifica**, em cada classe ou interface, as variáveis de instância e a assinatura completa dos métodos relevantes para a solução.

(c) **Explica**, de forma sucinta, a estrutura da solução e como ela tira partido do padrão escolhido.

NOME:

NUM:

Cada uma das 13 alíneas desta parte do teste contém uma frase inicial que é completada de cinco (5) formas diferentes. Indica, para cada um dos casos, se a frase inicial mais a parte que a completa é verdadeira (v) ou falsa (f). Cada resposta vale 0.15 valores: +0.15 val. se for escolhida a alternativa correta, -0.15 val. se for escolhida a alternativa incorreta. A cotação de cada alínea nunca é negativa, ou seja, no caso do número de respostas incorretas ser igual ou maior que o número de resposta corretas, a alínea é cotada com 0 (zero).

VERSÃO 1

[0.75 val.] 1. Os requisitos dum sistema de engenharia

- v ☒ f são equivalentes às necessidades dos utilizadores.
- ☒ v f incluem as restrições a que esse sistema está sujeito.
- v ☒ f têm que ser todos incorporados na solução para o sistema ter alta aceitação no mercado.
- ☒ v f representam características obrigatórias para satisfazer as necessidades dos utilizadores.
- ☒ v f podem corresponder a funcionalidades que um sistema deve possuir, para satisfação das necessidades da equipa de desenvolvimento (durante a manutenção).

[0.75 val.] 2. Um requisito funcional

- ☒ v f não deve depender de questões tecnológicas.
- v ☒ f diz-se implícito se for incluído na documentação pelos engenheiros.
- ☒ v f diz-se implícito se não tiver sido solicitado pelas partes interessadas.
- ☒ v f diz-se explícito se foi pedido pelos clientes e está incluído na documentação.
- ☒ v f pode ser afetado se forem considerados novos requisitos de desempenho.

[0.75 val.] 3. Os requisitos NÃO funcionais dum sistema

- ☒ v f correspondem a restrições que são impostas a esse sistema.
- v ☒ f modificam de forma muito significativa as funcionalidades desse sistema.
- v ☒ f tendem a ter um impacto em partes específicas desse sistema.
- v ☒ f têm pouco impacto na definição da sua arquitetura.
- v ☒ f não se influenciam uns aos outros.

[0.75 val.] 4. Os requisitos NÃO funcionais da categoria

- v ☒ f “disponibilidade” quantificam o tempo que o sistema demora a ser disponibilizado ao cliente.
- ☒ v f “usabilidade” estão relacionados com a facilidade de uso e de aprendizagem.
- v ☒ f “segurança” assumem duas vertentes principais: confidencialidade e autorização.
- v ☒ f “operacionais” referem-se a aspectos relacionados com a tecnologia que vai ser usada para construir o sistema.
- ☒ v f “operacionais” descrevem aspectos relacionados ao contexto em que o sistema vai funcionar.

[0.75 val.] 5. A engenharia de requisitos

- ☒ v f constitui uma das áreas de conhecimento do SWEBOK.
- ☒ v f é uma disciplina enquadrável na engenharia de sistemas.
- v ☒ f inclui sete atividades no seu processo: iniciação, levantamento, elaboração, negociação, documentação, validação e gestão.
- ☒ v f está focada na caracterização do problema que a equipa de desenvolvimento enfrenta.
- ☒ v f procura garantir que todos os requisitos relevantes serão conhecidos e compreendidos com o necessário nível de detalhe.

[0.75 val.] 6. O levantamento de requisitos

- ☒ v f permite identificar os requisitos que os utilizadores querem ver incorporados no sistema.
- ☒ v f deve poder ser realizado durante o ciclo de vida do sistema.
- ☒ v f pressupõe a escolha das técnicas a usar, pois cada uma delas é adequada a alguns contextos.
- v ☒ f inclui a negociação dos requisitos funcionais.
- ☒ v f inclui o estudo do domínio da solução.

Cotação: 1-13:

14:

15:

16:

TOTAL:

[0.75 val.] 7. As entrevistas

- ☒ v ☐ f podem ser realizadas com base num conjunto pré-definido de perguntas.
- ☐ v ☒ f podem ser usadas para produtos em massa, mesmo desconhecendo-se *a priori* os utilizadores.
- ☐ v ☒ f devem ser conduzidas de modo pouco regrado, para permitir descobrir aspetos inovadores.
- ☒ v ☐ f podem ser conduzidas por duas pessoas.
- ☒ v ☐ f devem ser realizadas com recurso a gravações áudio.

[0.75 val.] 8. Um modelo de domínio

- ☒ v ☐ f descreve o vocabulário e os conceitos relacionados com os sistemas que se podem desenvolver para o domínio considerado.
- ☐ v ☒ f descreve a arquitetura genérica dos sistemas dum dado domínio aplicacional.
- ☐ v ☒ f descreve apenas os aspetos comuns dos sistemas que se podem construir num dado domínio.
- ☒ v ☐ f expressa as verdades duradouras sobre o sistema que está em desenvolvimento.
- ☒ v ☐ f identifica e relaciona os conceitos do domínio a que diz respeito.

[0.75 val.] 9. A arquitetura dum sistema de software

- ☒ v ☐ f representa a visão macroscópica (i.e., o esqueleto) do sistema.
- ☒ v ☐ f descreve o conjunto de componentes e as relações que entre eles se estabelecem.
- ☐ v ☒ f é ortogonal aos requisitos funcionais.
- ☒ v ☐ f é especialmente crítica em sistemas de elevada complexidade.
- ☒ v ☐ f define a comunicação entre os seus componentes.

[0.75 val.] 10. Numa abordagem orientada ao risco,

- ☒ v ☐ f o esforço para conceber a arquitetura do sistema deve ser proporcional aos riscos do projeto.
- ☒ v ☐ f o esforço para conceber a arquitetura do sistema deve ser maior, se o domínio do projeto for novo (para a equipa).
- ☐ v ☒ f o risco é definido como a probabilidade de falha dividida pelo impacto dessa falha.
- ☐ v ☒ f a probabilidade de falha associada a um risco pode geralmente ser calculada com alta precisão.
- ☒ v ☐ f a arquitetura tem de ser cuidadosamente analisada, quando o espaço de soluções é reduzido

[0.75 val.] 11. Um padrão de conceção (design pattern)

- ☒ v ☐ f é um problema conhecido que tem uma solução bem definida.
- ☒ v ☐ f tipicamente envolve relações/interações entre classes ou objetos, no caso de se recorrer ao paradigma por objetos.
- ☒ v ☐ f permite a programadores juniores usar soluções propostas e testadas por especialistas.
- ☒ v ☐ f facilita a comunicação entre os engenheiros por serem usados conceitos padronizados.
- ☒ v ☐ f pode dar resultados indesejáveis, se aplicado num contexto inadequado.

[0.75 val.] 12. O padrão Observer

- ☐ v ☒ f não pode ser usado com outros padrões, para não pôr em causa a integridade do sistema.
- ☐ v ☒ f pressupõe que o modelo de dados possa ser “observer” da “view”.
- ☒ v ☐ f facilita a alteração da aparência e do comportamento da interface gráfica, mas implica normalmente modificar o modelo de dados.
- ☒ v ☐ f caracteriza-se por definir dependências “um para muitos” entre objetos.
- ☒ v ☐ f facilita a extensibilidade da solução quando são esperadas alterações nos observados.

[0.75 val.] 13. Um padrão arquitetural

- ☒ v ☐ f dá origem a sistemas mais fáceis de modificar.
- ☒ v ☐ f é aplicado antes dos padrões de conceção.
- ☒ v ☐ f implica a utilização de padrões de conceção.
- ☐ v ☒ f é obrigatoriamente especificado através de diagramas de classes UML.
- ☒ v ☐ f tem implicações em toda a estrutura de um sistema de software.

NOME:

NUM:

[3.5 val.] **14.** Considera as quatro frases F1-F4, que representam factos relevantes do processo de submissão e revisão de artigos científicos (numa revista de engenharia), que é suportado por um sistema informático. Cada pedido é submetido por um dos co-autores do artigo e revisto por um revisor indicado pelo editor da revista.

(a) Extrai de cada frase os **requisitos funcionais** que estão subjacentes e escreve-os segundo as recomendações dadas nas aulas. Indica, para cada um deles, se é requisito de utilizador ou de sistema.

(b) Identifica características não funcionais que estejam descritas nas frases.

- F1: Um dos co-autores preenche, através dum formulário desenvolvido em HTML/CSS, os dados relativos ao artigo (título, resumo, nomes dos autores, respetivas afiliações, etc) que pretende submeter.
- F2: A submissão deve incluir um ficheiro PDF, com um limite máximo de 35MB, que contém o artigo científico, escrito segunda as regras da revista.
- F3: O sistema deve alertar, através dum SMS, o editor para casos de artigos submetidos que tenham uma grande dose de sobreposição (plágio) nos conteúdos com artigos armazenados nos repositórios que foram previamente selecionados. O editor decide se cada artigo identificado com plágio está (ou não) apto a ser revisto.
- F4: O sistema deve ser fácil de usar, ter capacidade para ser acedido por 50 utilizadores ao mesmo tempo e disponibilizar um serviço de chat para esclarecimento de problemas técnicos e de dificuldades de utilização.

[3.25 val.] **15. (a) Elabora um modelo de domínio** para jogos de futebol de 11, considerando, pelo menos, os seguintes factos:

- uma partida é disputada numa determinada data e hora entre duas equipas e num dado campo/estádio de futebol
- cada equipa alinha em cada jogo com um mínimo de sete jogadores e um máximo de 11 jogadores
- cada equipa pode indicar até 7 suplentes
- cada equipa é orientada por um treinador ou por responsável técnico
- os jogadores de futebol são especializados na sua posição no campo: guarda-redes, defesas, meio-campistas e atacantes
- uma partida é regulada por um árbitro e dois árbitros assistentes
- uma partida dura 90 minutos, mas em alguns casos, um período extra de 30 minutos também é jogado

(b) Escreve um facto sobre jogos de futebol que possa ser acrescentado ao modelo de domínio. Introduz esse facto no modelo.

[3.5 val.] **16.** Pretende-se desenvolver um sistema de informação que permita a utentes de um serviço público de transportes receber informação em tempo real da posição e estimativa de chegada de um dado autocarro à sua próxima paragem. A informação a ser disponibilizada a diferentes tipos de clientes é composta pela posição (GPS) do veículo e pela estimativa do tempo de chegada (em minutos) à próxima paragem.

Conceptualiza um sistema que permita dar solução a este desafio, considerando que os autocarros são geradores desta informação e que é expectável a existência de diferentes tipos de clientes que consomem esta informação, e.g., utentes do serviço de transporte e gestores deste mesmo serviço. Os clientes poderão demonstrar interesse, e posterior desinteresse, em receber a informação a ser disponibilizada. Uma particularidade destes clientes é que poderão permitir a inclusão da opção de visualização desta informação em mapa.

Apresenta um diagrama UML para a possível solução, considerando um padrão adequado a este tipo de problema, para o caso particular em que existe um autocarro equipado com este sistema. Adicionalmente, deverão existir dois consumidores da informação: (1) na perspetiva do utente final (MobileClient) , e.g, smartphone de um gestor do serviço, (2) proveniente de uma aplicação web (WebClient). O WebClient deverá garantir a opção de visualização em mapa e o MobileClient não deverá incluir esta opção.

Indica as variáveis de instância e as assinaturas dos métodos relevantes para esta solução.