**Simulado de Engenharia de Software Ágil nro 03**

1. O diagrama de caso de uso da UML é uma ferramenta valiosa na fase de engenharia de requisitos. Seus elementos principais incluem atores, casos de uso e o limite do sistema. Qual das seguintes afirmações sobre a relação entre esses elementos é a mais correta?

1. Um caso de uso representa uma interação entre dois ou mais atores, e nunca com o sistema.
2. A única relação possível entre um ator e um caso de uso é a de generalização.
3. O limite do sistema define as fronteiras da aplicação, separando as funcionalidades que serão implementadas daquelas que não fazem parte do escopo do projeto.
4. O ator é sempre uma pessoa que utiliza o sistema, e o caso de uso é a funcionalidade que ela acessa.

2. Um sistema de e-commerce permite que um usuário faça o 'Cadastro de Cliente' e 'Faça Login' para acessar o 'Histórico de Compras'. A funcionalidade de 'Verificar Credenciais' é um pré-requisito obrigatório para o 'Histórico de Compras'. Qual das seguintes relações de caso de uso é a mais adequada para modelar esta dependência?

1. A funcionalidade 'Verificar Credenciais' deve usar a relação de inclusão (<<include>>) para se conectar ao caso de uso 'Histórico de Compras'.
2. O caso de uso 'Histórico de Compras' deve usar a relação de extensão (<<extend>>) para se conectar ao caso de uso 'Verificar Credenciais'.
3. O caso de uso 'Histórico de Compras' deve usar a relação de inclusão (<<include>>) para se conectar ao caso de uso 'Verificar Credenciais'.
4. O caso de uso 'Verificar Credenciais' deve usar a relação de extensão (<<extend>>) para se conectar ao caso de uso 'Histórico de Compras'.

3. Um sistema de gestão de biblioteca deve permitir que os usuários façam o 'Empréstimo de Livros', 'Renovem Empréstimos' e 'Devolvam Livros'. O sistema também deve ser capaz de 'Gerar Multa' para atrasos na devolução. Qual das seguintes modelagens de caso de uso representa a melhor prática para este cenário?

1. Modelar 'Empréstimo de Livros', 'Renovar Empréstimos' e 'Devolver Livros' como casos de uso separados e usar a relação de extensão (<<extend>>) do caso de uso 'Gerar Multa' para o caso de uso 'Devolver Livros' com a condição de atraso.
2. Modelar 'Devolver Livros' como um caso de uso e usar a relação de inclusão (<<include>>) para conectar ao caso de uso 'Gerar Multa'.
3. Modelar apenas 'Gerenciar Livros' como um único caso de uso para todas as funcionalidades.
4. Modelar 'Empréstimo de Livros', 'Renovar Empréstimos' e 'Devolver Livros' como casos de uso separados e usar a relação de extensão (<<extend>>) do caso de uso 'Devolver Livros' para o caso de uso 'Gerar Multa'.

4. Em um sistema de agendamento de consultas médicas, os seguintes elementos interagem com o software: 'Médico', 'Recepcionista', 'Paciente' e 'Sistema de Pagamento Externo'. Com base na definição de ator em UML, quais elementos seriam representados como atores no diagrama de caso de uso?

1. Apenas o 'Médico' e a 'Recepcionista', pois são as únicas pessoas que utilizam o sistema diretamente.
2. O 'Médico', a 'Recepcionista', o 'Paciente' e o 'Sistema de Pagamento Externo'.
3. Apenas o 'Paciente', pois é o usuário final que mais se beneficia das funcionalidades do sistema.
4. Apenas o 'Médico', a 'Recepcionista' e o 'Paciente', pois são os usuários humanos do sistema.

5. Uma empresa realizou um levantamento de requisitos de um Estacionamento, onde num primeiro momento destacou duas funcionalidades principais:

- Atendente registra a entrada e saída do veículo, mas é importante frisar que quando o cliente estaciona o veículo ele recebe o ticket onde contém a data e hora de entrada, placa, a cor do veículo e o modelo do carro.

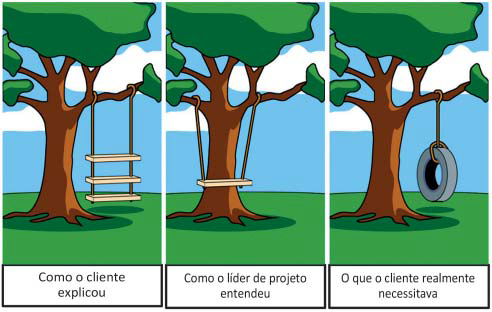
- Quando o cliente retira o veículo do estacionamento ele recebe o comprovante de pagamento (fatura).

É correto afirmar que:

1. Existe um relacionamento do tipo include do caso de uso Registrar Entrada para o caso de uso Gerar ticket impresso, onde este é essencial para o comportamento do caso de uso Registrar Entrada.
2. Existe um relacionamento do tipo include do caso de uso Gerar Fatura para o caso de uso Registrar Saída, onde este é opcional para o comportamento do caso de uso Registrar Entrada.
3. Existe um relacionamento do tipo generalização do caso de uso Gerar Fatura para o caso de uso registrar Saída, onde define uma funcionalidade do sistema do ponto de vista do usuário.
4. Existe um relacionamento do tipo extend do caso de uso Gerar Fatura para o caso de uso Registrar Saída, onde este é essencial para o comportamento do caso de uso Registrar Entrada.
5. Existe um relacionamento do tipo extend do caso de uso Registrar Entrada para o caso de uso Gerar ticket impresso, onde este é essencial para o comportamento do caso de uso Registrar Entrada.

6. Conforme ilustrado na figura apresentada, se os requisitos não forem levantados e documentados adequadamente, o projeto não terá sucesso. A inclusão de boas práticas na execução das fases da engenharia de requisitos (elicitação, análise e negociação, documentação, validação e gerenciamento) pode contribuir significativamente para a obtenção de um software que atenda as expectativas e necessidades do cliente.

SOMMERVILLE, I. A. N., RANSOM, J. *An Empírica/ Study of Industrial Requirements Engineering Process Assessment and lmprovement*. *Journal ACM Transactions on Software Engineering and Methodology*, 14(1), p.85-117, 2005 (com adaptações).



Disponível em <http://www.projectcartoon.com>. Acesso em 05 jul. 2017 (com adaptações).

Nesse contexto, as atividades referentes à fase de análise e negociação de requisitos são:

1. priorizar requisitos; classificar requisitos; planejar conflitos e resolvê-los.
2. criar um sumário de requisitos; definir termos especializados; rastrear requisitos.
3. registrar fonte de requisitos; procurar domínios de restrição; modelar casos de uso.
4. rastrear requisitos; definir políticas de mudança de requisitos; introduzir ferramentas de apoio.
5. um manual de usuário; organizar inspeções de requisitos; desenvolver caso de teste preliminar.

**Engenharia de requisitos**

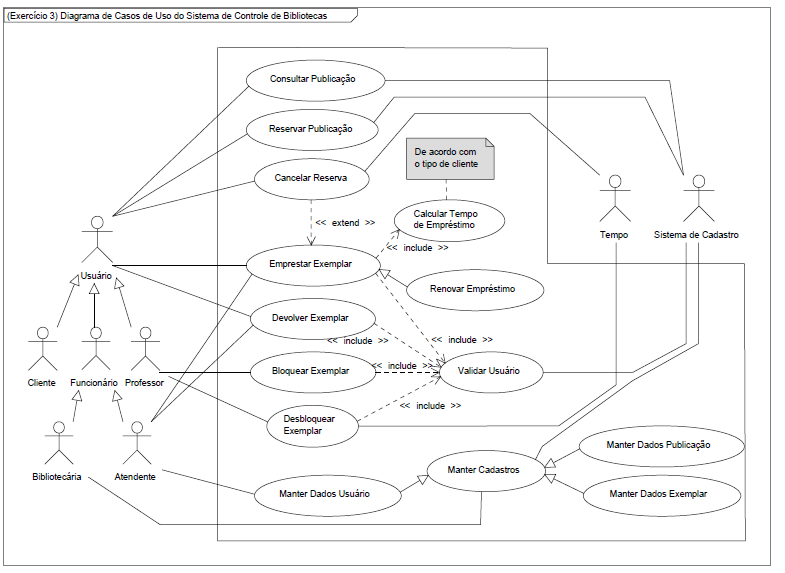
Grande parte das atividades realizadas em uma organização requer um planejamento prévio. Entre as diversas ações que podem ser tomadas durante a fase de planejamento para organizar uma atividade está a análise de requisitos.

O objetivo da análise de requisitos é definir necessidades, critérios e limitações para dados, sistemas ou processos que precisam ser implementados. Uma análise desse tipo gera um modelo de referência que pode ser consultado durante todo o projeto, inclusive durante a fase de validação, alinhando objetivos, necessidades e resultados.

A engenharia de requisitos é um ramo da engenharia de software que procura, de forma sistemática e repetitiva, obter, documentar e gerenciar requisitos para software que tenham qualidade e estejam alinhados com os objetivos da empresa. As fases da engenharia de requisitos estão explicadas a seguir.

* **Elicitação.** Fase que procura entender exatamente o problema que deve ser resolvido pelo software. Trata da procedência (origem) e da forma de coleta dos requisitos de software.
* **Análise e negociação**. Fase que compreende o detalhamento e a representação adequada e consistente dos requisitos recebidos durante a elicitação. O objetivo é ter mais entendimento da situação em que o problema acontece e o vislumbre prévio da solução.
* **Documentação.** Fase que diz respeito a todo o processo de criação de registros sobre o levantamento e a análise de requisitos. O objetivo é que tudo possa ser revisado e conferido em caso de necessidade.
* **Validação e gerenciamento**. Fase que considera a integridade dos requisitos levantados, garantindo que eles estejam consistentes e completos e que possam ser rastreados, mensurados, testados e até modificados, se houver necessidade. Também assegura que os requisitos não sejam ambíguos.

Observe o caso de uso abaixo e responda as questões 7, 8 e 9



7. Qual é o tipo de relacionamento entre os atores Cliente, Funcionário, Professor com o ator Usuário?

a) ( ) Inclusão

b) ( ) Exclusão

c) ( ) Herança

d) ( ) Generalização

e) ( ) Especialização

8. Os casos de uso Emprestar, Devolver, Bloquear, Desbloquear exemplar possui uma relação do tipo Inclusão com o caso de uso Validar usuário, isso significa que:

a) ( ) o caso de uso Validar usuário será executado, sempre que os demais casos de uso forem executados.

b) ( ) o caso de uso Validar usuário poderá ser executado, sempre que os demais casos de uso forem executados.

c) ( ) todos os caso de uso do domínio, dependem da execução do caso de uso Validar usuário.

d) ( ) ocorrerá uma generalização dos casos de usos para o caso de uso Validar usuário.

e) ( ) as execuções dos uses cases dependerão exclusivamente dos atores.

9. Em relação ao ator “Sistema de Cadastro”, assinale a alternativa correta:

a) ( ) está errado porque sistema não pode ser ator.

b) ( ) está certo porque ator pode só pode ser pessoa, e o único relacionamento possível é o de generalização.

c) ( ) está errado porque deveria ser um caso de uso do tipo extend.

d) ( ) está certo porque o ator especifica um papel executado por um usuário ou outro sistema que interage com o assunto (sistema).

e) ( ) está errado porque deveria ser um caso de uso do tipo include.

10. Os requisitos de usuário para um sistema devem descrever os requisitos funcionais e não funcionais de modo compreensível pelos usuários do sistema que não tem conhecimentos técnicos detalhados. Eles devem especificar somente o comportamento externo do sistema. Evitando tanto quanto possível as características de sistema. Os requisitos de sistema são as descrições mais detalhadas dos requisitos do usuário. Eles podem servir como base para um contrato destinado à implementação do sistema e, portanto, devem ser uma especificação completa e consistente de todo o sistema. Eles são utilizados pelos engenheiros de software como ponto de partida para o projeto do sistema. Tendo como base a Engenharia de Requisitos, assinale a alternativa **falsa**.

a) ( ) Requisitos não-funcionais são as qualidades globais de um software, como manutenibilidade, usabilidade, desempenho, custos e várias outras. Normalmente estes requisitos são descritos de maneira informal, de maneira controversa e são fáceis de validar.

b) ( ) O termo função é usado no sentido genérico de operação que pode ser realizada pelo sistema, seja através comandos dos usuários ou seja pela ocorrência de eventos internos ou externos ao sistema

c) ( ) Tradicionalmente, os requisitos de software são separados em requisitos funcionais e não-funcionais. Os requisitos funcionais são a descrição das diversas funções que clientes e usuários querem ou precisam que o software ofereça. Eles definem a funcionalidade desejada do software.

d) ( ) A especificação de um requisito funcional deve determinar o que se espera que o software faça, sem a preocupação de como ele faz. É importante diferenciar a atividade de especificar requisitos da atividade de especificação que ocorre durante o design do software.

e) ( ) No design do software deve-se tomar a decisão de quais funções o sistema efetivamente terá para satisfazer àquilo que os usuários querem.