

Análise do Sistema

Apresentação

Esta Unidade de Aprendizagem é dedicada à análise de um sistema, que é definida de modo geral como o estudo de um problema antes de passar à sua resolução.

Assim, nesta Unidade, você saberá como identificar um problema, quais as suas necessidades e objetivos, que interfaces são necessárias e que restrições devem ser consideradas no que diz respeito ao acesso do usuário.

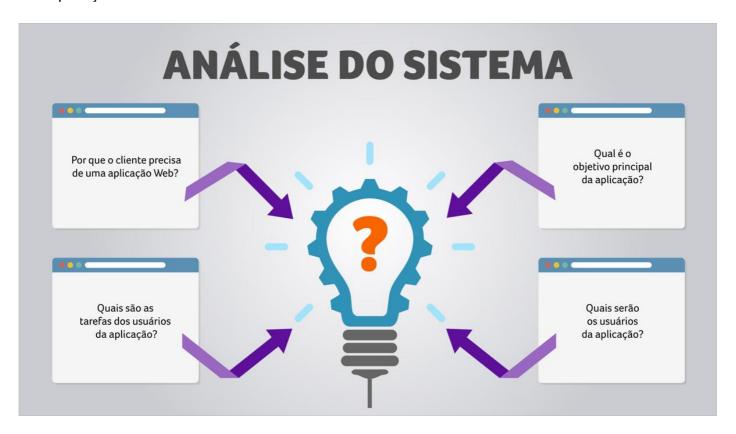
Bons estudos.

Ao final desta Unidade de Aprendizagem, você deve apresentar os seguintes aprendizados:

- Reconhecer as necessidades da aplicação.
- Identificar os objetivos da aplicação.
- Definir quais serão as tarefas dos usuários na aplicação.

Infográfico

O infográfico apresenta os questionamentos necessários antes de se iniciar o desenvolvimento de uma aplicação Web.



Conteúdo do Livro

Para que seja possível analisar um sistema, precisamos conhecer todas as suas funções, necessidades e objetivos. Somente com essas informações é que o desenvolvedor conseguirá identificar erros e falhas dentro do projeto. Para isso, estudar casos ou exemplos é uma ótima maneira de verificar essas questões.

Acompanhe um trecho da obra Desenvolvimento de software II. Inicie a leitura a partir do tópico Desenvolvimento de software para Web e finalize em Análise do sistema, em que será visto um estudo de caso.

Boa leitura.





D451 Desenvolvimento de software II [recurso eletrônico] : introdução ao desenvolvimento web com HTML, CSS, JavaScript e
PHP / Organizadores, Evandro Manara Miletto, Silvia de
Castro Bertagnolli. – Dados eletrônicos. – Porto Alegre :
Bookman, 2014.

Editado também como livro impresso em 2014. ISBN 978-85-8260-196-9

1. Informática – Desenvolvimento de software. 2. HTML. 3. CSS. 4. JavaScript. 5. PHP. I. Miletto, Evandro Manara. II. Bertagnolli, Silvia de Castro.

CDU 004.41

>>> Desenvolvimento de software para Web

A Internet introduziu, no cotidiano das pessoas, novos hábitos de entretenimento, comportamento, comunicação e consumo. O desenvolvimento de aplicações Web precisa acompanhar essas mudanças na rotina dos usuários, com a utilização de métodos e ferramentas específicas.

Conforme o direcionamento da aplicação Web, ela pode ser classificada de diferentes maneiras. A seguir estão elencadas algumas das principais classificações encontradas:

Site de conteúdo: apenas exibe um conteúdo de uma determinada área de forma linear, incluindo recursos (imagens, tabelas, gráficos, etc.) e navegação simples.

Site de registro de dados ou entrada do usuário: utiliza como recurso principal formulários que são preenchidos pelos usuários, com dados relacionados ao objetivo do site.

Portal: compreende um conjunto de páginas, links que convergem para um determinado contexto.

Aplicação orientada à transação: neste modelo, o usuário envia solicitações que a aplicação processa em um banco de dados, retornando ao usuário a resposta de sua solicitação.

As classificações não se restringem às descritas aqui. Uma lista mais completa pode ser encontrada em Pressman (2006) e em Sommerville (2011). Independentemente do tipo de aplicação Web a ser desenvolvida, sua qualidade pode ser determinada por diversos fatores, como: usabilidade, capacidade de navegação, possibilidade de busca e de recuperação, disponibilidade, tempo de resposta, processamento correto dos links (ligações entre páginas), segurança e uso de padrões.

Porém, para a obtenção de qualidade, é necessário utilizar um **processo sistema-tizado** para o desenvolvimento de sistemas Web. Esse processo deve prever que o prazo para a implantação do sistema seja mais rápido que o de outras aplicações. Ele deve prever etapas e recursos relacionados, como o projeto de interfaces gráficas com o usuário, que inclui aspectos como o layout. Observe que a qualidade visual do software é um dos fatores decisivos para o uso efetivo do sistema por usuários da Web (veja o Capítulo 3 para mais detalhes).

O processo também deve levar em consideração questões de segurança, pois uma aplicação Web está suscetível a ataques ou a acessos dos mais diferentes tipos de usuários. Desse modo, é necessário criar mecanismos para limitar o acesso a certos conteúdos e fornecer mecanismos seguros de transmissão dos dados (veja o Capítulo 10).

Além disso, o ideal é que o processo utilize um **modelo de desenvolvimento incremental**, o que significa dizer que todo o software é construído em partes. Cada uma dessas partes, quando entregue ao cliente, pode ser utilizada, e as partes previamente desenvolvidas são incorporadas, dando origem às versões do software.

O ponto de partida para a construção de uma aplicação Web é a identificação do **objetivo geral da aplicação**. Depois dessa etapa, é possível criar um modelo de análise, detalhando os requisitos (ver o tópico "Engenharia de requisitos" em Pressman, 2011, e Sommerville, 2011), como abordaremos no Capítulo 2. Na análise, serão identificados, além dos requisitos, os principais elementos de conteúdo das páginas.

Após compreender o propósito da aplicação, é necessário desenvolver os modelos de projeto (apresentados no Capítulo 2), bem como o projeto visual e gráfico (apresentado no Capítulo 3). Na etapa final, é encaminhada a **construção das páginas** com o uso de tecnologias específicas para a Web, como HTML, CSS, PHP, Javascript (Capítulos 4, 5, 7, 8 e 9) e ferramentas de edição de códigos.

À medida que cada funcionalidade é finalizada, deve ser testada. Assim, é necessário que o desenvolvedor verifique:

- Se a navegação está funcionando corretamente.
- Se existem erros na visualização do conteúdo da página, ou dos scripts utilizados por ela.
- Se os dados dos formulários estão sendo enviados corretamente ao servidor.
- Se o comportamento da aplicação é o mesmo em diferentes navegadores, visto que alguns possuem restrições com relação a recursos do HTML e CSS.



>> DICA

O modelo incremental gera como resultado um produto que pode ser utilizado ao término de cada incremento.



Agora é a sua vez!

Procure na Internet informações sobre os modelos de processo de software: sequencial linear, de prototipagem, RAD (Rapid Application Development), incremental e espiral. Liste as principais vantagens e desvantagens de cada um desses modelos e explique, com suas palavras, o porquê de o modelo incremental ser o ideal para o desenvolvimento de aplicações Web.

>>> Estudo de caso

O objetivo desta seção é apresentar um estudo de caso como uma "aplicação--guia" que será desenvolvida e aplicada em cada um dos capítulos que compõem este livro. Este estudo de caso estabelecerá soluções para uma situação-problema, com a finalidade de consolidar os aspectos teóricos, práticos e tecnológicos apresentados nos diferentes capítulos. Por meio do estudo de caso, você deve conseguir correlacionar as soluções apresentadas com outras situações práticas.

Para iniciar o desenvolvimento de uma aplicação Web, devemos responder a algumas questões:

- Por que o cliente precisa de uma aplicação para a Web?
- Qual é o objetivo principal da aplicação?
- Que processos ela deve controlar ou executar?
- Quais serão os usuários da aplicação?
- Quais são as tarefas dos usuários na aplicação?

Neste estudo de caso, o principal objetivo é criar uma aplicação Web que possibilite a venda de produtos no varejo pela Internet em uma loja virtual, ou seja, simular na Web alguns dos processos utilizados em uma loja física. Isso é necessário porque o cliente precisa ampliar sua área de atuação e expandir suas vendas.

O cliente analisou dados da Câmara Brasileira do Comércio Eletrônico (camara-e.net) e percebeu que os lucros com comércio eletrônico estão crescendo ano a ano e que algumas redes de varejo destacam percentuais de compras pela Internet iguais ou superiores aos das lojas físicas. Além disso, como sua loja vende produtos que estão nas categorias de produtos mais vendidos no e-commerce (produtos relacionados às áreas de cinema, fotografia, vídeo, áudio e de informática), o cliente resolveu contratar o desenvolvimento de uma aplicação Web.

Para utilizar a aplicação, o cliente acessa o site usando um navegador (Internet Explorer, Google Chrome ou Mozilla Firefox). Ao entrar na página principal, o sistema exibe como conteúdo as promoções do dia, opções de compra, busca, categorias de produtos e área de login e cadastro.

Com relação às categorias dos produtos a loja virtual vende:

- Na categoria "cinema e fotografia": câmeras, filmadoras, impressoras fotográficas, porta-retrato digital e acessórios.
- Na categoria "vídeo": TVs, Blu-Ray players, DVD players, suportes para TV, racks e acessórios.
- Na categoria "áudio": alguns tipos de aparelho de som (portátil, *mini-system*, *micro-system*), *home-theater*.
- Na categoria "informática": computadores portáteis, computadores *desktop*, impressoras, *scanners* e acessórios.

Caso o cliente deseje um produto específico, deve ser realizada uma pesquisa por meio do nome do produto, do fabricante, do preço ou de outras características que descrevem todos os produtos que se enquadram na mesma categoria. Outro ponto fundamental na exibição de produtos é a necessidade do sistema determinar, em tempo real, a disponibilidade dos produtos, exibindo somente aqueles que possuem quantidade superior a 1.

O resultado da pesquisa deve ser exibido no formato de uma lista ou grade, conforme o usuário achar mais conveniente. Todos os produtos que compõem a pesquisa podem ser ordenados por fabricante, nome e preço (ordem crescente ou decrescente). Se a pesquisa apresentar um número muito grande de produtos, o resultado deve ser organizado de modo que seja possível exibi-lo em páginas, para que o usuário possa selecionar a página que deseja visualizar.

Quando é selecionado um produto para a compra, o cliente da loja virtual deve ter um nome de usuário (o identificador utilizado é o endereço de e-mail) e senha válidos registrados no sistema. Assim, o acesso ao sistema é realizado somente por pessoas autorizadas, com as informações relacionadas às vendas acessadas somente pelo comprador e pelo funcionário encarregado de realizar a expedição do pedido.

Para registrar o usuário e a senha, é necessário que o cliente realize o cadastro, informando os seguintes dados: nome, e-mail, senha, CPF, telefone (residencial, comercial e celular) e, ainda, seu endereço (logradouro, número, complemento,

bairro, cidade, CEP, UF). Comumente, um cliente pode solicitar que suas compras sejam entregues em um endereço que não seja o residencial. Dessa forma, é necessário que vários endereços de entrega possam ser cadastrados.

O cliente pode sair do sistema por meio de um componente visual (botão ou link) disponibilizado na interface gráfica. Caso o login não seja encerrado, a sessão estabelecida pelo usuário será finalizada, automaticamente, em 20 minutos. Essa solução evita que o usuário esqueça o navegador aberto em seu computador e que outra pessoa efetue alguma compra usando dados já validados pela aplicação.

Um produto da lista, quando selecionado, deve ser exibido em outra página, que contém as principais informações. Caso o valor do produto seja diferente, conforme a forma de pagamento, essas informações devem ser apresentadas. Por exemplo, se o valor for um para o pagamento à vista ou com boleto, e outro para pagamento com cartão de crédito, isso deve ficar visível ao usuário.

Ainda, em conformidade com a legislação vigente no Brasil, caso incida juros sobre determinada forma de pagamento, é necessário exibir essas informações na tela do sistema. Nessa página, deve(m) constar a(s) foto(s) relacionada(s) com o produto, possibilitando a sua visualização em diferentes níveis de *zoom*. Para apresentar o valor final do produto, a página ainda deve solicitar o CEP do endereço de entrega do produto a fim de calcular o frete.

Cabe destacar que o sistema deverá estar integrado ao sistema dos Correios para determinar o estado da federação e o município onde o produto deverá ser entregue. Com isso, é possível gerar o valor correto do frete, e o usuário saberá exatamente o valor que pagará pelo(s) produto(s) incluído(s) no pedido.

Caso o usuário escolha um único produto, ele pode selecioná-lo e, após verificar as formas de pagamento e o valor do frete, comprá-lo. Ou pode selecionar vários produtos antes de finalizar a compra.

Para efetuar a compra do(s) produto(s), o usuário deve selecioná-lo(s) e incluí-lo(s) no carrinho de compras virtual. Esse carrinho contém a lista de todos os produtos selecionados para compra pelo cliente. Essa lista deve apresentar o nome do produto (e nesse nome deve existir um link para sua descrição), a quantidade desejada, o valor unitário e o valor total do produto (calculado com base no valor unitário e na quantidade de produtos definida).

Após, é necessário expor o subtotal do pedido e o valor do frete. Ao fim de todas essas informações, o sistema exibe o total do pedido, bem como o número de dias para a entrega. Geralmente, é utilizado o número de dias úteis para determinar o prazo de entrega no endereço solicitado. O sistema deve exibir os diversos endereços de entrega cadastrados para que o cliente escolha um, ou possibilitar que o cliente registre um endereço de entrega que ainda não existia no sistema.

O usuário, então, seleciona a forma de pagamento, que pode ser: cartão de crédito (com as bandeiras Visa, Mastercard, American Express, Diners Club International)

ou boleto. Caso o pagamento seja realizado com cartão, é necessário solicitar número, nome do titular, código de segurança e data de validade, bem como o CPF/CNPJ do titular do cartão. Esses dados servirão apenas para validação junto ao órgão financiador, não devendo ser armazenados no banco de dados do sistema para futuras consultas ou compras. Conforme o valor do produto, devem ser exibidas as formas de pagamento e as possibilidades de parcelamento do pagamento.

Se a opção for de pagamento por boleto, o sistema deve exibir, de forma clara e visível, diversas informações explicativas ao usuário, como:

- O valor total para pagamento à vista.
- A data-limite de pagamento do boleto.
- Que o valor é válido somente para a condição de pagamento com boleto bancário e em parcela única.
- Que o produto será expedido para o endereço de entrega somente após a confirmação de pagamento do boleto e que os prazos de entrega contam a partir da data de confirmação.
- As indicações de como imprimir o boleto e em qual tipo de papel e impressora o boleto deve ser impresso.
- Que o código de barras não pode ser furado, dobrado, amassado, rasurado ou riscado, pois isso impede o pagamento em qualquer agência bancária.

O sistema ainda deve prever a emissão de uma segunda via do boleto bancário, caso o usuário extravie ou danifique a primeira via. Também é necessário que o sistema possibilite ao usuário a utilização de cupons de desconto, que são aplicados sobre o valor total da venda. É importante observar que o valor do frete não recebe qualquer tipo de desconto.

Antes de concluir a compra, é necessário confirmar todos os dados com o cliente, para evitar inconsistências nos prazos, nas informações do pagamento e no endereço da entrega.

A aplicação deve emitir um comprovante do pedido realizado no sistema com os principais dados do(s) produto(s), bem como com detalhes de pagamento e prazo de entrega. Além disso, é informada ao cliente a situação de seu pedido, bem como o número para que ele possa, posteriormente, efetuar o acompanhamento do pedido pelo site.

Para o acompanhamento do pedido, o usuário deve realizar o login no sistema e acessar o item "pedidos". É exibida uma lista contendo todos os pedidos já realizados e a situação de cada um deles. Ao selecionar os detalhes de um pedido, o sistema exibe a data/hora da compra, os produtos comprados, o endereço indicado para a entrega, as formas de pagamento e a situação. No caso da situação, é



>> DEFINIÇÃO Modelo lógico é um modelo utilizado para mapear as entidades que fazem parte da solução de um problema. Ele define as chaves (primária e estrangeira), segue as regras de normalização e de integridade referencial, entre outros conceitos vinculados a banco de

dados.

possível saber se ele já foi expedido ou está em transporte, bem como a previsão de entrega na casa do cliente.

Algumas características importantes desse sistema se apresentam quando o pagamento é efetivado, como:

- Para realizar a baixa do(s) produto(s) no estoque, o sistema deve se comunicar com o sistema de estoque.
- Para realizar a expedição do(s) produto(s), é necessário emitir a nota fiscal (que possui um número, uma relação dos produtos vendidos e os dados essenciais do comprador).

A Figura 1.4 esquematiza um modelo lógico inicial que foi sugerido como base para o desenvolvimento do estudo de caso apresentado no livro *Desenvolvimento de software I: conceitos básicos* (OKUYAMA; MILETTO; NICOLAO, 2014). Observe que, para mapear a realidade apresentada pelo estudo de caso descrito previamente, será necessário criar alguns diagramas e acrescentar entidades a esse modelo. O Capítulo 2 descreverá alguns passos e diagramas que você poderá utilizar para fazer o mapeamento correto dos requisitos e do banco de dados.



>> NO SITE

Você pode usar várias ferramentas para criar modelos lógicos de bancos de dados. No caso deste livro, os modelos lógicos serão todos criados com a ferramenta gratuita BRmodelo, disponível no ambiente virtual de aprendizagem Tekne.

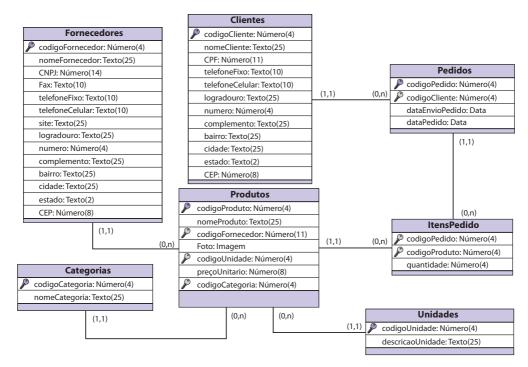


Figura 1.4 Modelo lógico inicial.

Fonte: dos autores.



>>> Agora é a sua vez!

Refaça o modelo lógico apresentado na Figura 1.4 com a ferramenta BRmodelo. Analise o enunciado do estudo de caso e verifique quais informações devem ser acrescentadas/retiradas do diagrama para que ele fique coerente.

REFERÊNCIAS

OKUYAMA, F. Y.; MILETTO, E. M.; NICOLAO, M. *Desenvolvimento de software I*: conceitos básicos. Porto Alegre: Bookman, 2014.

PRESSMAN, R. S. Engenharia de software: uma abordagem profissional. 7. ed. Porto Alegre: AMGH, 2011.

SOMMERVILLE, I. Engenharia de software. 9. ed. São Paulo: Pearson Education, 2011.

LEITURAS RECOMENDADAS

COULOURIS, G. et al. Sistemas distribuídos: conceitos e projetos. 5. ed. Porto Alegre: Bookman, 2013.

PFLEEGER, S. L. Engenharia de software: teoria e prática. 2. ed. São Paulo: Prentice Hall, 2003.

TANENBAUM, A. S.; STEEN, M. V. Sistema distribuídos. 2. ed. São Paulo: Prentice-Hall, 2007.

>>> Delimitação do sistema

A delimitação do sistema consiste em determinar o objetivo geral da aplicação e algumas de suas metas por meio de questionamentos que auxiliam a compreender melhor a aplicabilidade do sistema que está sendo desenvolvido.

Vamos começar o desenvolvimento identificando o objetivo geral da aplicação e respondendo aos questionamentos apresentados no Quadro 2.1. Para exemplificar cada questionamento, bem como sua aplicabilidade, foi utilizado o estudo de caso descrito no Capítulo 1.

Após responder esses questionamentos, você pode ainda agendar algumas reuniões com o cliente para verificar os seguintes itens:

- Já existe um sistema semelhante na empresa? Se sim, quem o utiliza?
- Quais são as outras pessoas da empresa que podem dar informações adicionais sobre o problema ou sobre os processos que o sistema terá que solucionar?

Quadro 2.1 » Exemplo de delimitação do problema		
Questionamento	Resposta	Fundamentação
Por que o cliente precisa de uma aplicação para a Web?	Para aumentar o volume de vendas de sua loja.	É utilizado para que a aplicação se concentre em atender a principal necessidade do cliente, que, em nosso estu- do de caso, é aumentar o volume de vendas da loja do cliente.
Qual é o principal objetivo da aplicação?	Venda de produtos pela Internet.	Tem a função de delimitar o objetivo geral da aplicação Web sem que os detalhes sejam descritos. Analisando o estudo de caso, chegamos ao objetivo principal do sistema "Vender produtos pela Internet".
Que processos a aplica- ção deve controlar ou executar?	Venda de produtos e ge- renciamento de pedidos.	Propicia a percepção sobre qual é o núcleo do sistema, ou seja, quais são as atividades que devem ser priorizadas no desenvolvimento incremental. Considerando a aplicação proposta, deduz-se que o núcleo do sistema é a venda de produtos e o gerenciamento de pedidos, em que gerenciar consiste em acompanhar pedidos (cliente) e expedir pedidos (funcionário).
Quais serão os usuários da aplicação?	Clientes que compram os produtos e funcionários que realizarão a expedição dos produtos comprados. Ambos os usuários utilizam um navegador.	É identificado quem utilizará o sistema para, posterior- mente, estabelecer o perfil do usuário, ou seja, determi- nar como ele irá interagir a partir de seus conhecimentos prévios. Basicamente, o estudo de caso terá dois usuá- rios (cliente e funcionário), que realizarão atividades di- ferentes no sistema.
Quais são as tarefas dos usuários da aplicação?	Compra de produtos, acompanhamento de pedidos e expedição de pedidos.	Permite identificar as atividades principais sem descrever quais passos são necessários para realizá-las. Considerando o nosso exemplo, para comprar um produto o usuário necessita realizar uma pesquisa ou comprar produtos que estão na "capa" do site em oferta.

Com essas perguntas, é possível detectar problemas frequentes no desenvolvimento de software – como, por exemplo, usuários resistentes a mudanças – ou sistemas semelhantes que você poderá utilizar como fonte de informações para o desenvolvimento do seu sistema.

Além disso, fazendo os questionamentos mencionados, você consegue verificar a viabilidade de desenvolvimento do sistema, pois saberá quais são as funcionalidades que deverão ser atendidas, qual é a plataforma do sistema (desktop, cliente-servidor, móvel, etc.), se haverá restrições de segurança, de desempenho, de usabilidade, entre outros pontos. Você ainda poderá analisar se sua equipe possui condições técnicas de desenvolver a aplicação ou se terá que aprender a



tecnologia antes de criar a solução para o cliente, pois isso impacta no prazo para realização do projeto.

Com base em tudo o que foi explicado anteriormente, podemos delimitar o escopo da aplicação, pois já sabemos o que vamos desenvolver e qual é o objetivo principal do sistema. A próxima seção irá apresentar alguns aspectos teórico-práticos vinculados à análise de um sistema.

>> DICA

Nunca estime prazos para a entrega de um projeto sem consultar sua equipe ou analisar muito bem o que o sistema deve fazer. Você pode perder a confiança do cliente se combinar um prazo de entrega e postergar esse prazo várias vezes. Se você não conhece técnicas de estimativa de tempo, pesquise ou use o PMBOK como guia.



Agora é a sua vez!

- 1. Monte o Quadro 2.1, com exceção da coluna "Fundamentação", para os sistemas descritos abaixo:
 - Locadora de carros
 - Escola de idiomas
 - · Venda de passagens aéreas
- 2. Se um cliente solicitasse que você desenvolvesse os sistemas acima para a plataforma Web, você mudaria alguma coisa na delimitação de cada um dos problemas?



>>> Análise do sistema

Um dos pontos principais da análise do sistema é descobrir o que deve ser feito, sem focar em como fazer. Assim, para iniciar a etapa de análise, é necessário compreender o problema sem se preocupar com como cada funcionalidade vai ser desenvolvida.



» DEFINIÇÃO

Um **requisito** é uma condição ou uma capacidade com a qual o sistema deve estar de acordo.

Após delimitar o sistema, você pode começar a identificar os requisitos da aplicação. Essa tarefa é uma etapa essencial para desenvolver um software, pois com ela podemos identificar as necessidades do usuário e traduzi-las em funcionalidades da aplicação.

Os requisitos podem ser classificados conforme a sua funcionalidade:

Funcionais: estão relacionados diretamente com as funcionalidades que a aplicação deve oferecer aos usuários. Exemplo: cadastro de usuários.

Não funcionais: têm relação com os requisitos de qualidade e de segurança, que se caracterizam por impor limitações na aplicação. Exemplo: disponibilidade, autenticação de usuário (*login*), sistema para Internet.

Quando falamos em requisitos, você pode pensar que encontrá-los é algo muito trivial. Porém, se pesquisarmos mais a fundo, veremos o quanto é difícil identificar os requisitos e modelá-los de forma que todos os envolvidos entendam o que deve ser feito. Se você pesquisar na Internet por "técnicas para encontrar requisitos", encontrará uma lista bem extensa.

Uma das técnicas que costuma se destacar é a de **revisar requisitos existentes**. Essa técnica consiste na análise de especificações de requisitos de sistemas já utilizados pelo cliente ou de outros sistemas com objetivo principal semelhante. Por exemplo, para construir a aplicação da loja virtual de nosso estudo de caso, você pode analisar diversas lojas virtuais, identificar as principais funcionalidades e verificar se elas são adequadas à aplicação que está desenvolvendo. Muitas vezes, é possível identificar funcionalidades que o cliente não havia previsto ou, até mesmo, que havia esquecido.

À medida que você vai descobrindo os requisitos, pode ir esquematizando o que se espera da aplicação por meio de um diagrama ou modelo. Se você pesquisar, verá que o diagrama de casos de uso da Unified Modeling Language (UML) é uma notação muito utilizada para mapear as funcionalidades de uma aplicação.

As próximas seções vão apresentar como criar um diagrama de casos da UML com a utilização do estudo de caso.

Dica do Professor

Assista ao vídeo sobre a análise do sistema e os modelos de processo de software.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Exercícios

1)	São características do modelo incremental:
A)	Possui gerenciamento de riscos, mas de difícil atualização.
B)	Possui gerenciamento de riscos e não é flexível.
C)	É flexível, possui gerenciamento de riscos e de fácil atualização.
D)	É flexível, mas não possui gerenciamento de riscos.
E)	Não é flexível, não possui gerenciamento de riscos e é de difícil atualização.
2)	Qual a função do banco de dados em uma aplicação Web?
A)	O banco de dados não é necessário.
B)	É onde as informações ficam salvas, mas não se tem acesso a elas.
C)	Apenas manter informações para consulta, não permitindo gravação.
D)	Formatar as cores e fontes dos textos.
E)	É onde as informações ficam salvas e podem ser lidas e alteradas.
3)	O que é um modelo lógico?
A)	Modelo utilizado para o design da interface da aplicação.
B)	Modelo utilizado para mapear as partes da solução de um problema.
C)	Modelo utilizado para tornar mais fácil a experiência de utilização da aplicação pelo usuário
D)	Modelo que define quais linguagens de programação serão utilizadas na aplicação.
E)	Modelo que define o nível de acesso do usuário.

O que é uma chave primária?

B)	A principal tabela do banco de dados.
C)	A coluna das senhas no banco de dados.
D)	A coluna que é utilizada como índice no banco de dados.
E)	Uma senha provisória gerada automaticamente para o usuário acessar pela primeira vez a aplicação.
5)	O que é uma chave estrangeira?
A)	O mesmo que chave primária.
B)	A coluna na mesma tabela que faz relação à chave primária.
C)	A chave primária de outro banco de dados.
D)	A coluna em outra tabela que faz relação à chave primária.
E)	A coluna que fica com os dados vazios para uma futura utilização.

A) A senha de acesso ao banco de dados.

Na prática

Uma loja especializada na venda de sapatos masculinos contratou uma empresa para criar um site de vendas *online* responsivo. Cláudio, desenvolvedor da empresa contratada, reuniu-se com o responsável da loja de sapatos para entender por que o cliente precisava da aplicação, quais os objetivos, qual o perfil dos usuários que utilizam a aplicação e quais são as tarefas desses futuros usuários. Com as respostas anotadas, Cláudio iniciou o processo de análise do sistema para avaliar a implantação de um site responsivo. Só assim o desenvolvedor pode identificar erros e falhar dentro do projeto.



Saiba mais

Para ampliar o seu conhecimento a respeito desse assunto, veja abaixo as sugestões do professor:

Como identificar necessidades dos consumidores?



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Problemas no processo de desenvolvimento de software



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Engenharia de software

Conteúdo interativo disponível na plataforma de ensino!