

Olá aluno(a). Vamos falar sobre requisitos de software?

### **Você sabe o que é um requisito?**

Segundo o dicionário Oxford requisito é uma condição para se alcançar determinado fim.

### **E na área de TI?**

Eles definem o que o sistema deve fazer, como deve se comportar e quais são as suas características fundamentais. Os requisitos formam a base para o projeto, desenvolvimento, teste e validação de um software.

### **Qual é a importância dos requisitos em um projeto de software:**

1. Requisitos claros e bem definidos permitem que todas as partes interessadas compreendam as metas e funcionalidades do projeto. Isso evita mal-entendidos e garante que todos tenham a mesma visão do que está sendo desenvolvido.
2. Os requisitos capturam as necessidades reais dos usuários e clientes. Isso ajuda a garantir que o software entregue realmente o que os usuários esperam, resultando em maior satisfação.
3. O design do sistema é influenciado pelos requisitos. A arquitetura, interfaces e estruturas são moldadas para atender às especificações dos requisitos.
4. Os requisitos fornecem critérios para testes. Eles ajudam os testadores a verificar se o software está funcionando conforme o esperado e se atende às necessidades dos usuários.
5. Requisitos bem documentados servem como base para rastrear e gerenciar mudanças ao longo do ciclo de vida do projeto. Isso ajuda a evitar a adição de novos requisitos de forma arbitrária e a manter a consistência.

6. Requisitos detalhados permitem estimativas mais precisas de tempo e recursos necessários para o desenvolvimento. Isso ajuda no planejamento do projeto.
7. Requisitos bem definidos e compreendidos desde o início reduzem a probabilidade de retrabalho, o que pode resultar em economia de tempo e dinheiro.
8. Ter requisitos claros ajuda a estabelecer expectativas realistas entre as partes interessadas. Isso evita surpresas e desapontamentos ao longo do desenvolvimento.

Em resumo, os requisitos são a base para o sucesso de um projeto de desenvolvimento de software. Eles asseguram que todas as partes envolvidas tenham uma visão compartilhada do sistema a ser construído e garantem que o software atenda aos objetivos e às necessidades dos usuários finais.

## **PARA PRATICAR**

Considere a seguinte situação:

Uma cadeia de hotéis de luxo deseja desenvolver um sistema de gerenciamento de reservas para melhorar a eficiência e a experiência do cliente. Atualmente, eles gerenciam as reservas manualmente, o que resulta em atrasos, erros e insatisfação dos clientes. O novo sistema deve ser capaz de lidar com reservas, check-ins, checkouts e pagamentos de forma integrada.

A partir desse cenário realize o levantamento de requisitos funcionais e não funcionais de cada situação a seguir. Cada cenário oferece um contexto único para o levantamento de requisitos funcionais e não funcionais, permitindo que você analise as necessidades e expectativas dos diferentes stakeholders.

### **Cenário 1: Reserva Perdida**

Maria e Carlos vão fazer uma viagem comemorativa. Eles decidiram reservar um quarto em um hotel de luxo. No entanto, ao tentarem fazer a reserva por telefone, o atendente cometeu um erro ao inserir as datas. Quando chegaram ao hotel, descobriram que não havia quartos disponíveis para as datas corretas. Maria e Carlos ficaram frustrados com a situação e desejaram que o sistema de reservas pudesse garantir a precisão e a disponibilidade das datas.

### **Cenário 2: Check-In**

Fernanda é uma executiva que viaja frequentemente a negócios. Ela chegou ao hotel após um longo voo e estava ansiosa para fazer o check-in rapidamente. No entanto, a fila na recepção estava longa, e o processo de check-in manual estava demorando consideravelmente. Fernanda ficou impaciente e percebeu que a eficiência do check-in era crucial, especialmente para viajantes cansados. Ela imaginou como seria mais conveniente se o hotel tivesse um sistema automatizado para agilizar o processo.

### **Cenário 3: Fatura Detalhada**

Paulo e Marta, estavam prestes a fazer o check-out após uma estadia agradável. No entanto, quando receberam a fatura no momento do check-out, ficaram surpresos com a falta de detalhes. Não estava claro quais serviços adicionais eles haviam solicitado e como esses serviços foram cobrados. Paulo e Marta desejavam que o sistema de gerenciamento de reservas pudesse gerar faturas detalhadas para que eles pudessem revisar e entender facilmente os custos.

### **Cenário 4: Pagamento Fácil**

Ricardo, um empresário ocupado, estava prestes a finalizar sua estadia no hotel. No entanto, ao tentar fazer o pagamento, ele percebeu que o sistema não aceitava seu cartão de crédito. Ele teve que fornecer os detalhes manualmente, o que foi inconveniente e aumentou o risco de erro humano. Ricardo desejou

que o processo de pagamento fosse simplificado e que o sistema pudesse processar pagamentos de forma segura e eficaz.

### **Cenário 5: Acessibilidade**

André, um hóspede com deficiência visual, estava animado para desfrutar de sua estadia no hotel de luxo. No entanto, ele enfrentou dificuldades para interagir com o sistema de reserva, que não era acessível para pessoas com deficiência visual. André percebeu a importância de ter um sistema que fosse compatível com tecnologias assistivas, permitindo a inclusão de todos os hóspedes.

### **GABARITO**

Lembro que essas são sugestões de resposta. Você pode descrever os requisitos de forma diferente ou identificar outros que não estão no gabarito.

### **Cenário 1: Reserva Perdida**

#### Requisitos Funcionais:

1. RF1: O sistema deve verificar automaticamente a disponibilidade de quartos para as datas selecionadas antes de finalizar a reserva.
2. RF2: O sistema deve alertar o usuário caso não haja quartos disponíveis para as datas desejadas e oferecer opções alternativas.
3. RF3: Deve ser possível que os usuários editem as datas da reserva antes da confirmação final.

#### Requisitos Não Funcionais:

1. RNF1: O sistema deve ter um tempo de resposta rápido para verificar a disponibilidade dos quartos (abaixo de 3 segundos).

2. RNF2: A interface do sistema deve ser intuitiva e fácil de entender para evitar erros de reserva.

## **Cenário 2: Check-In**

### Requisitos Funcionais:

1. RF1: O sistema deve permitir o check-in online, permitindo que os hóspedes forneçam detalhes antecipadamente.
2. RF2: Deve ser possível realizar o check-in na recepção usando quiosques de autoatendimento.
3. RF3: O sistema deve notificar os hóspedes quando seus quartos estiverem prontos para o check-in.

### Requisitos Não Funcionais:

1. RNF1: O tempo de espera na fila de check-in deve ser reduzido para menos de 5 minutos.
2. RNF2: Os quiosques de autoatendimento devem ser de fácil utilização, mesmo para hóspedes não familiarizados com tecnologia.

## **Cenário 3: Fatura Detalhada**

### Requisitos Funcionais:

1. RF1: O sistema deve gerar automaticamente faturas detalhadas no momento do check-out.
2. RF2: As faturas devem listar todos os serviços extras solicitados pelos hóspedes durante a estadia.

### Requisitos Não Funcionais:

1. RNF1: As faturas devem ser enviadas aos hóspedes por e-mail em um formato legível e de fácil compreensão.

2. RNF2: O sistema deve garantir a precisão das informações na fatura para evitar disputas.

#### **Cenário 4: Pagamento Fácil**

##### Requisitos Funcionais:

1. RF1: O sistema deve aceitar uma variedade de métodos de pagamento, incluindo cartões de crédito.
2. RF2: Os usuários devem poder salvar os detalhes do cartão de crédito para pagamentos futuros.
3. RF3: O sistema deve fornecer uma confirmação imediata após o pagamento ser processado.

##### Requisitos Não Funcionais:

1. RNF1: O sistema deve ser compatível com os padrões de segurança de pagamento para proteger as informações do cartão de crédito dos hóspedes.
2. RNF2: Os pagamentos devem ser processados em tempo real para evitar atrasos nas confirmações.

#### **Cenário 5: Acessibilidade**

##### Requisitos Funcionais:

1. RF1: O sistema deve ser compatível com tecnologias inclusivas, como leitores de tela.
2. RF2: A interface do sistema deve ser projetada com contraste adequado e fontes legíveis para facilitar a leitura.

##### Requisitos Não Funcionais:

1. RNF1: O sistema deve seguir as diretrizes de acessibilidade, para garantir a inclusão de pessoas com deficiência.

2. RNF2: Testes de usabilidade devem ser realizados com usuários com deficiência para validar a eficácia das melhorias de acessibilidade.

Cada levantamento de requisitos aborda as necessidades específicas de cada cenário, tanto em termos de funcionalidades quanto de considerações não funcionais. A seguir você encontrará um exemplo de documento de requisitos para o cenário apresentado.

## **Documento de Requisitos - Sistema de Gerenciamento de Reservas de Hotel**

### **1. Introdução**

O Sistema de Gerenciamento de Reservas de Hotel visa melhorar a eficiência e a experiência do cliente para uma cadeia de hotéis de luxo. Atualmente, as reservas são gerenciadas manualmente, resultando em atrasos, erros e insatisfação dos clientes. O novo sistema será uma solução integrada para lidar com reservas, check-ins, check-outs e pagamentos de forma eficiente.

### **2. Requisitos Funcionais**

#### **2.1. Reservas Online**

1. O sistema deve permitir que os clientes visualizem a disponibilidade de quartos para datas específicas.
2. Os clientes devem ser capazes de selecionar o tipo de quarto desejado, número de hóspedes e datas de check-in e check-out.
3. O sistema deve exibir as tarifas dos quartos e calcular o custo total da estadia.

## **2.2. Check-in e Check-out**

1. A equipe do hotel deve ser capaz de realizar o check-in dos hóspedes, registrando suas informações pessoais.
2. O sistema deve permitir o registro dos detalhes de pagamento no momento do check-in.
3. Os hóspedes devem receber um e-mail de confirmação após o check-in bem-sucedido.

## **2.3. Serviços Adicionais**

1. Os hóspedes devem poder solicitar serviços adicionais, como serviço de quarto ou transporte para o aeroporto.
2. O sistema deve registrar as solicitações de serviços adicionais e enviar notificações à equipe relevante.

## **2.4. Faturas Detalhadas**

1. O sistema deve gerar automaticamente faturas detalhadas no momento do check-out.
2. As faturas devem listar todos os serviços extras solicitados pelos hóspedes durante a estadia.

# **3. Requisitos Não Funcionais**

## **3.1. Desempenho**

1. O sistema deve ter um tempo de resposta médio de menos de 3 segundos para verificar a disponibilidade de quartos.
2. A latência para confirmações de reserva não deve exceder 2 segundos.



### **3.2. Usabilidade**

1. A interface do usuário deve ser intuitiva e de fácil navegação para os clientes e a equipe do hotel.
2. A equipe do hotel deve receber treinamento adequado para usar o sistema sem dificuldades.

### **3.3. Segurança**

1. O sistema deve cumprir os padrões de segurança PCI-DSS para proteger os dados do cartão de crédito dos hóspedes.
2. O acesso às informações dos hóspedes deve ser restrito apenas a pessoal autorizado.

### **3.4. Escalabilidade**

1. O sistema deve ser projetado para lidar com pelo menos 1000 reservas por dia.
2. Deve ser possível adicionar novos hotéis à cadeia sem afetar a funcionalidade do sistema.

## **4. Rastreabilidade de Requisitos**

Cada requisito funcional e não funcional deve ser rastreável a partir do documento de requisitos até a fase de implementação e testes.

## **5. Assinaturas**

Este documento foi revisado e aprovado por:

Nome: \_\_\_\_\_

Cargo: \_\_\_\_\_

Data: \_\_\_\_\_

## **6. Versão do Documento**

Versão: 1.0

Data de Revisão: [data de revisão]

Responsável pela Revisão: [nome do responsável]

Alterações: [descreva as alterações feitas nesta versão]

Este é um exemplo simplificado de um documento de requisitos para o cenário do hotel. Em um ambiente de desenvolvimento real, o documento seria mais detalhado e incluiria mais seções, como diagramas de casos de uso, requisitos específicos de cada funcionalidade e muito mais.

Lembre-se que a elaboração de requisitos não é apenas uma etapa preliminar, mas um processo iterativo contínuo que se estende ao longo de todo o ciclo de desenvolvimento. À medida que o software evolui, os requisitos também podem evoluir, adaptando-se a novas necessidades, feedback dos usuários e mudanças no ambiente.