[ÂNIMA - UNIVERSIDADE POTIGUAR](https://analisederequisitos.com.br)

[PROJETO](https://analisederequisitos.com.br/termo-de-abertura-do-projeto-exemplo) DE MODELAGEM DE SOFTWARE

**SISTEMA DE RESERVA DE HÓTEIS**

NATAL/2025

Preparado por

Janio Garcia Ramalho Leite(CCOMP)

Luis Guilherme Avelino de Araújo Santos (ADS)

Gabriel Henrique Alves e Souza(ADS)

Jardson Ribeiro Costa(CCOMP)

Eduardo Araújo Guedes   
  
MATUTINO

DOCUMENTO M-0027

ÚLTIMA ATUALIZAÇÃO: 06/05/2025

Sumário

[1. Introdução 3](#_Toc164617430)

[Escopo do Produto 3](#_Toc164617431)

[Especificações 3](#_Toc164617432)

[2. Convenções, termos e abreviações 3](#_Toc164617433)

[Identificação dos Requisitos 3](#_Toc164617434)

[Prioridades dos Requisitos 3](#_Toc164617435)

[3. Descrição geral do sistema 4](#_Toc164617436)

[Abrangência e sistemas relacionados 4](#_Toc164617437)

[Descrição dos usuários 4](#_Toc164617438)

[4. Requisitos funcionais 4](#_Toc164617439)

[Casos de Uso 5](#_Toc164617440)

[5. Requisitos não funcionais 5](#_Toc164617441)

[Usabilidade 6](#_Toc164617442)

[Confiabilidade 6](#_Toc164617443)

[Desempenho 6](#_Toc164617444)

[Segurança 6](#_Toc164617445)

[6. Artefatos UML 7](#_Toc164617446)

[7. Banco de Dados 7](#_Toc164617447)

[8. <Opcional> Descrição da interface com o usuário 7](#_Toc164617448)

[9. CONCLUSÕES E RECOMENDAÇÕES 8](#_Toc164617449)

[10. REFERÊNCIAS 8](#_Toc164617450)

Histórico de Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **DATA** | **VERSÃO** | **DESCRIÇÃO DA ALTERAÇÃO** | **RESPONSÁVEL** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Você deverá ler esse documento completo. Todo o texto explicativo existente para explicar a redação desse documento deve ser excluído, inclusive esse aqui. É importante a leitura do documento para entender todas as informações necessárias para construir o documento de requisitos de software que deve ser o mais informativo e detalhado possível pois, este será a planta onde os programadores irão construir o sistema.   
  
A tabela acima é o histórico de revisão conforme o padrão doa ISO 9000. O Histórico somente deve ser alterado quando uma vez que esse documento tenha sido concluído, ocorra alguma alteração no mesmo. Não deve ser atualizado enquanto se está trabalhando no documento.

# Introdução

O **Sistema de Reserva de Hotéis** destina-se a automatizar o processo de gerenciamento de reservas em hotéis de alto padrão, permitindo que clientes e funcionários realizem operações de cadastro, consulta, reserva, pagamento e cancelamento de forma integrada e segura. O caráter transdisciplinar deste projeto engloba as Unidades Curriculares de Modelagem de Software e Programação e Soluções Computacionais, atendendo aos critérios de ambas as disciplinas.

# Escopo do Produto

* **O que o sistema faz:**
  + Cadastro e realiza autenticação de clientes e funcionários.
  + Consulta de disponibilidade de quartos por data.
  + Seleção de serviços extras (café da manhã, translado, lavanderia).
  + Registro de reservas, processamento de pagamento(simulação) e atualização de status de quartos.
  + Cancelamento de reservas com reversão de disponibilidade.
* **O que o sistema não faz:**
  + Emissão de notas fiscais eletrônicas.
  + Integração com sistemas de fidelidade ou programas de milhagem.
  + Integração com Gateway de Pagamento.

# Especificações do Hardware

* **REQUISITOS MINIMOS:**
* CPU: Processador Intel Core i3 (4 núcleos, 3.0 GHz) ou equivalente AMD Ryzen 3
* **Memória RAM**: 4 GB DDR4
* **Armazenamento**: 500 MB de espaço livre em disco (quando em SSD, tempo de acesso otimizado)
* **Rede**: Conexão de 10 Mbps de banda mínima

REQUISITOS RECOMENDADOS:

* **CPU**: Intel Core i5 (6 núcleos, 3.5 GHz) ou equivalente AMD Ryzen 5
* **Memória RAM**: 8 GB DDR4 ou superior
* **Armazenamento**: SSD de 20 GB para sistema e banco de dados local
* **Rede**: Conexão de 100 Mbps

# Abrangência e sistemas relacionados

ESCOPO POSITIVO:

Cadastro e autenticação de perfis. Consulta, seleção e reserva de quartos. Serviços extras e cálculo de custo. Processamento de pagamento e logs. Cancelamento e histórico completo.

ESCOPO NEGATIVO:

Emissão de NFe — ERP Fiscal externo. Programa de fidelidade — módulo futuro. Relatórios de BI avançados — ferramenta externa.

# Descrição dos usuários

6.1 Perfil Comum Usuários com conhecimento básico em desktop e navegação.

6.2 Cliente Função: reserva de quartos e serviços.

Necessidades: rapidez, clareza, validações.

Desafios: conflitos de datas, entendimento de custos.

6.3 Funcionário Função: confirmação, modificação e cancelamento.

Necessidades: visão consolidada, filtros.

Desafios: manter disponibilidade consistente.

6.4 Administrador Função: configuração de quartos, serviços, usuários.

Necessidades: logs, relatórios mínimos.

Desafios: integridade e auditabilidade.

# Requisitos funcionais

<Nesta seção, apresente todos os requisitos funcionais e casos de uso do sistema. Em sistemas grandes é comum haver muitos requisitos e casos de uso e, para facilitar a visualização deste documento, você pode agrupá-los em subseções de casos de uso correlacionados. Os nomes das subseções devem ser únicos e pequenos (3 palavras no máximo) e podem ser formados por palavras, números e/ou abreviações.

Cada um dos casos de uso deve ser descrito em um bloco específico, seguindo o modelo descrito abaixo. O identificador do bloco deve conter o número do caso de uso (por exemplo, [RF001]) e o seu nome. Se os casos de uso forem agrupados em subseções específicas, a numeração deles deve ser reiniciada a cada subseção (dentro de uma mesma subseção, todo caso de uso deve ter um número de identificação único).>

**Identificação dos Requisitos**

Por convenção, a referência a requisitos é feita através do nome da subseção onde eles estão descritos, seguido do identificador do requisito, de acordo com o esquema abaixo: [nome da subseção. Identificador do requisito]

Por exemplo, o requisito funcional “Recuperação de dados” está descrito em uma subseção em um bloco identificado pelo número [RF016] da seguinte forma: “[RF016] - Recuperação de dados”. Já o requisito não funcional “Confiabilidade” está descrito na seção de requisitos não funcionais em um bloco identificado por [NF008] da seguinte forma: “[NF008] – Confiabilidade”.

**Prioridades dos Requisitos**

Para estabelecer a prioridade dos requisitos foram adotadas as denominações “essencial”, “importante” e “desejável”.

• **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.

• **Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.

• **Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

### **[RF001] <Nome do Requisito>**

<Opcional – forneça uma pequena explicação do propósito do caso de uso (útil quando o nome do caso de uso não deixa suficientemente claro qual é o seu objetivo) e o(s) seu(s) respectivo(s) ator(es). Em seguida, substitua um dos símbolos abaixo por , para indicar a prioridade do caso de uso.>

**Ator: <informe o(s) ator(es) do caso de uso >**

**Prioridade**:  Essencial  Importante  Desejável

### **[RF…] <Nome do Requisito >**

<Utilize os mesmos campos mostrados no bloco anterior para descrever este e os demais requisitos funcionais (casos de uso) desta subseção.>

# Casos de Uso

usecase "Acessa o site" as C1

usecase "Realiza o cadastro" as C2

usecase "Seleciona uma data" as C3

usecase "Realiza reserva" as C4

usecase "Seleciona serviços adicionais" as C5

usecase "Realiza pagamento" as C6

usecase "Recebe notificação" as C7

usecase "cancelar reserva" as C9

usecase "É direcionado a uma mesa disponível" as C10

usecase "Escolhe sua refeição" as C12

usecase "Pede a conta" as C14

usecase "Direciona o cliente a uma mesa disponível" as R1

usecase "Registra o pedido" as R3

usecase "Emite a conta do cliente" as R6

usecase "Recepciona o cliente na chegada" as UC1

usecase "Valida os documentos da reserva" as UC3

usecase "Realiza o check-in" as UC4

usecase "Instrui sobre funcionamento e comodidades" as UC7

usecase "Entrega a chave de acesso ao quarto" as UC8

usecase "Realiza o check-out" as UC10

usecase "Registar consumo restaurante" as UC11

usecase "Registra pagamento do cliente" as UC12

usecase "Agradecer pela preferência na hospedagem" as UC13

usecase "Entrega o cartao de acesso do quarto" as UC14

usecase "Gerar relatorios" as G1

usecase "Gerenciar reservas" as G2

usecase "Interditar quarto" as G3

usecase "Realocar quarto" as G4

usecase "Ocupação, receita, etc" as G5

usecase "Comunica a equipe de limpeza ou de reforma" as G6

usecase "Enviar confirmação por email" as S2

usecase "Validar dados do cliente" as S3

usecase "Gerar número de reserva" as S4

usecase "Gerar notificações" as S5

usecase "Calcular valor total da reserva" as S8

usecase "Enviar confirmação por email" as S9

usecase "Validar dados do Cliente" as S10

usecase "Solicita lavanderia, café da manha e garagem" as CL11

usecase "Calcular valor total da reserva" as CL12

usecase "Realiza o Login" as ex1

usecase "Escolhe pagar conta como consumo restaurante" as ex2

# Requisitos não funcionais

#### Usabilidade

### **RNF01 – Aprendizado Rápido Usuários novatos devem ser capazes de completar uma reserva básica em, no máximo, 3 minutos sem necessidade de treinamento prévio. Prioridade: Essencial**

**RNF02 – Consistência de Interface**  
Elementos de interface (botões, menus, campos de formulário) devem seguir padrões visuais e posicionais uniformes em todas as telas, garantindo previsibilidade.  
**Prioridade:** Importante

**RNF03 – Acessibilidade**  
A interface deve estar em conformidade com WCAG 2.1 AA, suportando leitores de tela, navegação por teclado e contraste mínimo de 4.5:1.  
**Prioridade:** Essencial

#### Confiabilidade

**RNF04 – Disponibilidade**  
O sistema deve estar disponível no mínimo 99,9 % do tempo (não incluindo janelas de manutenção programada).  
**Prioridade:** Essencial

**RNF05 – Tolerância a Falhas**  
Em caso de falha de um componente (servidor, serviço ou banco), o sistema deve continuar operando por meio de redundância, sem interrupção perceptível para o usuário.  
**Prioridade:** Essencial

**RNF06 – Monitoramento e Alerta**  
Deve haver monitoramento em tempo real de saúde dos componentes, com alertas automáticos (e-mail/SMS) para a equipe de TI sempre que métricas críticas ultrapassarem limiares configuráveis.  
**Prioridade:** Importante

#### Desempenho

**RNF07 – Tempo de Resposta**

* Sob carga normal, cada página ou operação crítica (busca de quartos, submissão de reserva, confirmação de pagamento) deve responder em no máximo 1 segundo.
* **Prioridade:** Essencial

**RNF08– Usuários Concorrentes**

* Deve suportar pelo menos 500 usuários autenticados navegando e interagindo de forma concorrente, mantendo os tempos de resposta conforme o RNF23.
* **Prioridade:** Essencial

**RNF09 – Escalabilidade**

* A arquitetura deve permitir adição de nós (servers) para atender ao dobro de carga atual sem necessidade de reengenharia significativa.
* **Prioridade:** Desejável

#### Segurança

**RNF10– Autenticação Forte**

* O sistema deve exigir senhas com ao menos 8 caracteres, incluindo letras maiúsculas, minúsculas, números e símbolos, e suportar autenticação multifator (MFA).
* **Prioridade:** Essencial

**RNF12 – Controle de Acesso / Autorização**

* Deve implementar controle de acesso baseado em funções (RBAC), de modo que cada usuário só veja e execute as operações previstas para seu perfil (cliente, recepcionista, gerente, administrador)
* **Prioridade:** Essencial

# Artefatos UML

<Apresentar as imagens com os artefatos UML, bem como os diagramas de os Casos de Uso, Diagrama de Entidade e Relacionamento, além de outros diagramas se necessários.>

### **Diagrama de Caso de Uso**

Interface gráfica do usuário, Diagrama

O conteúdo gerado por IA pode estar incorreto.

### **Diagrama de Classe**

Diagrama

O conteúdo gerado por IA pode estar incorreto.

### **Diagrama de Sequência**

Tabela

O conteúdo gerado por IA pode estar incorreto.

### **Diagrama Entidade Relacionamento - DER**

### **Outros diagramas**

# Banco de Dados

CREATE TABLE Clientes (

id INT auto\_increment primary key,

nome VARCHAR(100) NOT NULL,

senha VARCHAR(255) NOT NULL,

cpf VARCHAR(14) NOT NULL UNIQUE,

telefone VARCHAR(20) NOT NULL,

email VARCHAR(70) NOT NULL UNIQUE,

data\_nascimento DATE DEFAULT NULL

);

CREATE TABLE Funcionario (

id INT auto\_increment primary key,

nome VARCHAR(100) NOT NULL,

cargo VARCHAR(30) NOT NULL,

login VARCHAR(255) NOT NULL UNIQUE,

senha VARCHAR(255) NOT NULL

);

CREATE TABLE Quartos (

id INT AUTO\_INCREMENT PRIMARY KEY,

numero INT NOT NULL UNIQUE,

tipo VARCHAR(50) NOT NULL,

situacao VARCHAR(20) NOT NULL,

valor\_diaria DECIMAL(10,2) NOT NULL

);

CREATE TABLE Reservas (

id INT AUTO\_INCREMENT PRIMARY KEY,

id\_quarto INT NOT NULL,

id\_cliente INT NOT NULL,

data\_checkin DATE NOT NULL,

data\_checkout DATE NOT NULL,

valor\_total DECIMAL(10,2) NOT NULL,

FOREIGN KEY (id\_quarto) REFERENCES Quartos(id),

FOREIGN KEY (id\_cliente) REFERENCES Clientes(id)

);

CREATE TABLE Pagamentos (

id INT AUTO\_INCREMENT PRIMARY KEY,

id\_reserva INT NOT NULL,

valor DECIMAL(10,2) NOT NULL,

data\_pagamento DATE NOT NULL,

metodo ENUM('CartaoCredito', 'CartaoDebito', 'Dinheiro', 'PIX', 'Boleto') NOT NULL,

FOREIGN KEY (id\_reserva) REFERENCES Reservas(id)

);

CREATE TABLE categorias\_quartos (

id INT auto\_increment PRIMARY KEY ,

nome VARCHAR(50) NOT NULL ,

tipo\_quarto ENUM( 'standard', 'luxo', 'suite', 'vip') NOT NULL DEFAULT 'standard' ,

descricao TEXT ,

preco\_base DECIMAL(10,2) NOT NULL

);

CREATE TABLE Servicos\_Adicionais (

id INT AUTO\_INCREMENT PRIMARY KEY,

nome VARCHAR(100) NOT NULL UNIQUE,

descricao VARCHAR(255) DEFAULT NULL,

valor DECIMAL(10,2) NOT NULL

);

CREATE TABLE Cardapio (

id INT AUTO\_INCREMENT PRIMARY KEY,

nome VARCHAR(100) NOT NULL UNIQUE,

descricao VARCHAR(255) DEFAULT NULL,

valor DECIMAL(10,2) NOT NULL

);

CREATE TABLE Conta\_Restaurante (

id INT AUTO\_INCREMENT PRIMARY KEY,

id\_reserva INT NOT NULL,

data\_pedido DATE NOT NULL,

valor\_total DECIMAL(10,2) NOT NULL,

FOREIGN KEY (id\_reserva) REFERENCES Reservas(id)

);

INSERT INTO Clientes (nome, senha, cpf, telefone, email, data\_nascimento) VALUES

('Joao', 'Joao123', '123.456.789-00', '(11)98765-4321', 'joao@gmail.com', '1985-04-12'),

('Maria', 'Maria123', '987.654.321-00', '(21)91234-5678', 'maria@gmail.com', '1990-07-23'),

('Lucas', 'Lucas123', '111.222.333-44', '(31)99888-7766', 'lucas@gmail.com', '1988-05-16'),

('Ana', 'Ana123', '555.666.777-88', '(41)99977-6655', 'ana@gmail.com', '1992-08-05'),

('Pedro', 'Pedro123', '999.888.777-66', '(51)97766-5544', 'pedro@gmail.com', '1995-11-30');

INSERT INTO Funcionario (nome, cargo, login, senha) VALUES

('Carlos', 'Recepcionista', 'carlos', 'Carlos1234'),

('Ana', 'Gerente', 'ana', 'Ana1234'),

('Paulo', 'Limpeza', 'paulo', 'Paulo1234'),

('Sofia', 'Cozinha', 'sofia', 'Sofia1234'),

('Rafael', 'Seguranca', 'rafael', 'Rafael1234');

INSERT INTO categorias\_quartos (nome, tipo\_quarto, descricao, preco\_base) VALUES

('Standard', 'standard', 'Quarto confortável para até 2 pessoas', 150.00),

('Luxo', 'luxo', 'Quarto com vista para o mar e varanda', 300.00),

('Suíte', 'suite', 'Suíte espaçosa com sala de estar', 450.00),

('VIP', 'vip', 'Quarto com serviço exclusivo e jacuzzi', 700.00),

('Economico', 'standard', 'Quarto simples e econômico', 100.00);

INSERT INTO Quartos (numero, tipo, situacao, valor\_diaria) VALUES

(101, 'standard', 'disponivel', 150.00),

(102, 'luxo', 'ocupado', 300.00),

(201, 'suite', 'disponivel', 450.00),

(301, 'vip', 'manutencao', 700.00),

(103, 'standard', 'disponivel', 140.00);

-- Agora o id dos quartos será 1, 2, 3, 4, 5 respectivamente

INSERT INTO Reservas (id\_quarto, id\_cliente, data\_checkin, data\_checkout, valor\_total) VALUES

(1, 1, '2025-06-10', '2025-06-15', 750.00),

(3, 2, '2025-07-01', '2025-07-05', 1800.00),

(5, 3, '2025-06-20', '2025-06-22', 280.00),

(2, 4, '2025-08-10', '2025-08-15', 1500.00),

(4, 5, '2025-09-01', '2025-09-05', 3500.00);

INSERT INTO Pagamentos (id\_reserva, valor, data\_pagamento, metodo) VALUES

(1, 750.00, '2025-06-01', 'PIX'),

(2, 1800.00, '2025-06-20', 'CartaoCredito'),

(3, 280.00, '2025-06-19', 'Dinheiro'),

(4, 1500.00, '2025-08-01', 'Boleto'),

(5, 3500.00, '2025-08-25', 'CartaoDebito');

INSERT INTO Servicos\_Adicionais (nome, descricao, valor) VALUES

('Lavanderia', 'Serviço de lavagem e passagem de roupas', 50.00),

('Translado Aeroporto', 'Transporte de ida e volta ao aeroporto', 100.00),

('Cafe da Manha', 'Café da manhã servido no quarto', 30.00),

('Estacionamento', 'Vaga coberta para o carro', 40.00),

('Wi-Fi Premium', 'Internet de alta velocidade no quarto', 20.00);

INSERT INTO Cardapio (nome, descricao, valor) VALUES

('Hamburguer', 'Hamburguer com queijo e bacon', 25.00),

('Salada Caesar', 'Salada com alface, croutons e molho especial', 20.00),

('Suco Natural', 'Suco de laranja natural', 10.00),

('Pizza Portuguesa', 'Pizza com presunto, ovos e azeitonas', 40.00),

('Cafe Expresso', 'Cafe expresso tradicional', 8.00);

INSERT INTO Conta\_Restaurante (id\_reserva, data\_pedido, valor\_total) VALUES

(1, '2025-06-11', 55.00),

(2, '2025-07-02', 35.00),

(3, '2025-06-21', 60.00),

(4, '2025-08-12', 25.00),

(5, '2025-09-03', 45.00);

# <Opcional> Descrição da interface com o usuário

<Esta seção deve conter desenhos, mockups, ou rascunhos das telas do sistema que forem necessários ou convenientes para esclarecer algum dos requisitos do sistema. Para sistemas que possuem protótipos ou versões já desenvolvidas é possível capturar as telas e apresentar figuras delas.

### **[INT001] <Identificador de uma interface>**

<Descreva a interface em questão, através de figuras, diagramas e/ou texto.>

### **<Opcional> Críticas da interface**

<Você pode fazer aqui a descrição de críticas simples de interface, como o tamanho e máscara de campos, simplificando assim a descrição dos fluxos de exceção.>

### **[INT002] <Identificador de outra interface>**

<Prossiga no detalhamento das interfaces do sistema, descrevendo todas que for necessário, cada uma em uma subseção.>

# CONCLUSÕES E RECOMENDAÇÕES

<Um breve texto com as conclusões de todo o trabalho. Deve citar os principais pontos observados e descrever um breve relato explicitando como o aluno e/ou grupo interpreta estes pontos>.

# REFERÊNCIAS

<Para melhor realização desse trabalho, foram indicadas leituras de e-books, textos, aulas, em roteiros de estudos predefinidos, mas caberá aos alunos, e/ou grupos, o empenho em pesquisar nas bibliotecas da Universidade, na Biblioteca virtual ou em outras fontes por iniciativa própria do aluno e/ou grupo>