



**Ramiro  
Zavala**

# **Engenharia de Requisitos**

**Caso : Aplicativo para Petshop**



# Sumário

• Introdução	<b>01</b>
• Requisitos funcionais e não funcionais	<b>02</b>
• O documento de requisitos de software	<b>03</b>
• Especificação de requisitos	<b>04</b>
• Processos de Engenharia de Requisitos	<b>05</b>
• Elicitação e Análise de requisitos	<b>06</b>
• Validação de requisitos	<b>07</b>
• Gerenciamento de requisitos	<b>08</b>
• Histórias de usuário	<b>09</b>
• Casos de uso BDD e TDD	<b>10</b>
• Conclusão	<b>11</b>



# INTRODUÇÃO

Criar um aplicativo de gerenciamento de PETSHOP envolve compreender e documentar detalhadamente as necessidades dos clientes e dos usuários finais. A engenharia de requisitos é a prática que nos ajuda a fazer isso. Neste e-book, vamos explorar como aplicar esses conceitos para desenvolver um aplicativo de petshop, incluindo a definição de requisitos funcionais e não funcionais, histórias de usuário, e a aplicação de BDD (Behavior Driven Development) e TDD (Test Driven Development).



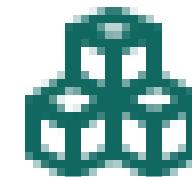
2

# REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS



# Requisitos

## Para um aplicativo de PetShop



### Funcionais

- Cadastro de Clientes e Pets.
- Agendamento de serviços (banho, tosa, consulta veterinária).
- Gerenciamento de estoque de produtos.
- Processamento de pagamentos.
- Emissão de recibos e notas fiscais.
- Comunicação com clientes via e-mail e mensagens.



### Não Funcionais

- Segurança para proteger dados sensíveis.
- Desempenho para garantir resposta rápida.
- Facilidade de uso.
- Conformidade com leis e regulamentos.

# O DOCUMENTO DE REQUISITOS DE SOFTWARE

O documento de requisitos de software detalha todas as funcionalidades e características do aplicativo. Este documento inclui descrições, fluxogramas, diagramas de caso de uso, etc.

# ESPECIFICAÇÃO DE REQUISITOS

A especificação de requisitos transforma os requisitos em descrições detalhadas para os desenvolvedores. Isso pode incluir : Diagramas UML para visualização, Lista detalhada de funcionalidades, Protótipos, etc.

# Exemplo de especificação de requisitos

Para o requisito funcional de Agendamento de serviços

## Descrição

.....● O sistema deve permitir que o usuário agende serviços para Pets, selecionando o tipo de serviço , data e hora.

## Fluxograma

.....● Mostra o processo de seleção do serviço, escolha de Data e Hora, e Confirmação.

5

# PROCESSOS DE ENGENHARIA DE REQUISITOS



# Etapas



6

# ELICITAÇÃO E ANÁLISE DE REQUISITOS





### Elicitação de Requisitos

- Entrevistas.
- Questionários.
- Workshops.



### Análise de Requisitos

- Garantir a clareza e consistência dos requisitos,
- Resolver conflitos.



### Exemplo : Agendar Consulta

- Elicitar : Entrevistas com veterinários, funcionários e clientes.
- Analisar : Verificar que os requisitos sejam claros e não contraditórios.

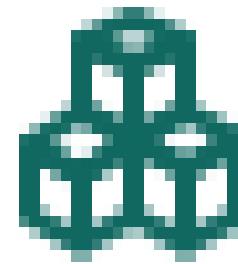
7

# VALIDAÇÃO DE REQUISITOS





## Métodos de Validação



- Revisões por pares.
- Protótipos.
- Testes de conceito.

## Exemplo : Controle de Estoque



- **Validação** : Criação de protótipos para testar a adição de novos itens ao estoque.

# GERENCIAMENTO DE REQUISITOS





## Importância

- Acompanhar as mudanças nos requisitos.
- Garantir que todos estejam cientes das mudanças.



## Ferramentas

- Software de gerenciamento de requisitos.
- Comunicação regular com os Stakeholders.

# Análise de Ponto de Função

## Definição

É uma técnica de estimativa de software que mede a funcionalidade fornecida ao usuário final, baseada em seus requisitos.

## Componentes

- Entradas Externas : EE
- Saídas Externas : SE
- Consultas Externas : CE
- Arquivos Lógicos Internos : ALI
- Arquivos de Interface Externa : AIE

## Exemplo de Análise de Ponto de Função Para aplicativo de PETSHOP

Componente	Descrição	Contagem
Entradas Externas (EE)	Formulário de cadastro de Cliente	4 PF
Entradas Externas (EE)	Formulário de cadastro de Pet	4 PF
Saídas Externas (SE)	Relatório de Agendamentos	5 PF
Consultas Externas (CE)	Consulta de disponibilidade de serviços	3 PF
Arquivos Lógicos Internos (ALI)	Banco de dados de Clientes e Pets	7 PF
Arquivos de interface Externa (AIE)	Integração com sistema de pagamentos	6 PF

**TOTAL : 29 Pontos de Função**

# Matriz de Rastreabilidade

## Definição

É uma ferramenta que ajuda a rastrear os requisitos ao longo do ciclo de vida do projeto. Ela garante que todos os requisitos sejam cobertos pelos testes e que as mudanças possam ser facilmente gerenciadas.

## Exemplo de Matriz de Rastreabilidade

Para aplicativo de PETSHOP

Requisito	Casos de Uso	BDD / TDD
Cadastro de Clientes e Pets	UC01 : Cadastro de Cliente UC02 : Cadastro de Pet	Teste 1.1 : Cadastro de novo Cliente Teste 2.1 : Cadastro de novo Pet
Agendamento de serviços	UC03 : Agendar serviço	Teste 3.1 : Agendamento de banho Teste 3.2 : Cancelamento de agendamento
Gerenciamento de estoque	UC04 : Adicionar produto	Teste 4.1 : Adiciona novo produto Teste 4.2 : Alerta de baixa de estoque
Processamento de pagamentos	UC05 : Processar pagamento	Teste 5.1 : Pagamento de serviço Teste 5.2 : Relatório de pagamentos

9

# HISTÓRIAS DE usuário



# Histórias de Usuário

Para os requisitos funcionais do aplicativo de Petshop

01

## Cadastro de Clientes e Pets

- Como cliente, quero criar um perfil no aplicativo para armazenar minhas informações pessoais e dos meus pets.
- Como funcionário, quero cadastrar novos clientes e seus pets, para manter o registro atualizado.
- Como administrador, quero editar e remover cadastros de clientes e pets, para manter a base de dados limpa.
- Como cliente, quero ver o histórico dos serviços prestados aos meus pets, para acompanhar sua saúde e bem-estar.

02

## Agendamento de Serviços

- Como cliente, quero agendar um banho para meu pet, para garantir que ele esteja sempre limpo.
- Como funcionário, quero visualizar todos os agendamentos do dia para organizar meu trabalho.
- Como cliente, quero cancelar um agendamento com antecedência para que o Petshop possa se reprogramar.
- Como administrador, quero gerenciar horários disponíveis para otimizar a agenda do Petshop.

03

## Gerenciamento de Estoque

- Como administrador, quero adicionar novos produtos ao estoque para manter o inventário atualizado.
- Como funcionário, quero verificar a quantidade de produtos em estoque para atender os pedidos dos clientes.
- Como administrador, quero receber alertas de baixa de estoque para reabastecer antes que os produtos acabem.
- Como funcionário, quero registrar a venda de produtos para manter o controle de saída de estoque.

# Histórias de Usuário

Para os requisitos funcionais do aplicativo de Petshop

04

## Processamento de Pagamentos

- Como cliente, quero pagar pelos serviços do meu pet de forma rápida e segura.
- Como administrador, quero gerar relatórios de pagamentos para monitorar as finanças do Petshop.
- Como cliente, quero ter várias opções de pagamento (cartão, dinheiro, pix) para escolher a mais conveniente.
- Como funcionário, quero confirmar o recebimento dos pagamentos para concluir o atendimento.

05

## Emissão de Recibos e NF

- Como cliente, quero receber um recibo após cada pagamento para ter um comprovante.
- Como administrador, quer emitir notas fiscais eletrônicas para conformidade tributária.
- Como cliente, quero a opção de receber meus recibos por e-mail para facilitar o armazenamento.
- Como administrador, quero visualizar e imprimir recibos antigos para consultas futuras.

06

## Comunicação com Clientes

- Como cliente, quero receber lembretes dos meus agendamentos para não esquecer.
- Como administrador, quero enviar promoções e novidades para todos os clientes, para aumentar as vendas.
- Como cliente, quero receber notificações de saúde e bem-estar dos meus pets para estar sempre informado.
- Como funcionário, quero enviar mensagens personalizadas aos clientes para fortalecer o relacionamento.

10

# CASOS DE USO BDD e TDD



# Casos de Uso BDD

## Cadastro de Clientes e Pets

**Cenário 1**

### Cadastro de novo cliente

- Dado que um novo cliente deseja se cadastrar
- Quando ele preenche o formulário de cadastro com suas informações
- Então o sistema deve criar um novo perfil de cliente

**Cenário 2**

### Cadastro de novo pet

- Dado que um cliente deseja cadastrar seu pet
- Quando ele preenche o formulário com as informações do pet
- Então o sistema deve criar um novo registro de pet

## Agendamento de Serviços

**Cenário 1**

### Agendamento de banho

- Dado que um cliente deseja agendar um banho para seu pet
- Quando ele escolhe a data, hora e confirma o agendamento
- Então o sistema deve registrar o agendamento no calendario

**Cenário 2**

### Cancelamento de Agendamento

- Dado que um cliente deseja cancelar um agendamento
- Quando ele solicita o cancelamento antes da data agendada
- Então o sistema deve remover o agendamento no calendario

# Casos de Uso BDD

## Gerenciamento de Estoque de Produtos

**Cenário 1**

### Adição de novos produtos

- Dado que um administrador deseja adicionar novos produtos ao estoque
- Quando ele insere os dados do produto e confirma
- Então o sistema deve atualizar o inventário com o novo produto

**Cenário 2**

### Alerta de baixa de estoque

- Dado que um produto está com baixa quantidade no estoque
- Quando a quantidade disponível atinge o limite mínimo
- Então o sistema deve enviar um alerta de baixa de estoque ao administrador

## Processamento de Pagamentos

**Cenário 1**

### Pagamento de serviços

- Dado que um cliente deseja pagar por um serviço
- Quando ele escolhe o método de pagamento e conclui a transação
- Então o sistema deve processar o pagamento e registrar a transação

**Cenário 2**

### Relatórios de pagamento

- Dado que um administrador deseja visualizar os pagamentos recebidos
- Quando ele gera um relatório de pagamentos
- Então o sistema deve exibir um relatório detalhado com todas as transações

# Casos de Uso BDD

## Emissão de Recibos e Notas Fiscais

**Cenário 1**

### Emissão de Recibo

- Dado que um cliente concluiu um pagamento
- Quando o pagamento é confirmado
- Então o sistema deve emitir um recibo e disponibilizá-lo ao cliente

**Cenário 2**

### Envio de Recibo por e-mail

- Dado que um cliente prefere receber recibos digitalmente
- Quando ele opta por receber por e-mail
- Então o sistema deve enviar o recibo para o e-mail do cliente

## Comunicação com Clientes

**Cenário 1**

### Notificação de lembrete de agendamento

- Dado que um agendamento está próximo
- Quando a data do serviço se aproxima
- Então o sistema deve enviar uma notificação de lembrete ao cliente

**Cenário 2**

### Envio de promoções

- Dado que o Petshop tem uma nova promoção
- Quando o administrador cria a campanha promocional
- Então o sistema deve enviar a promoção para todos os clientes

# Casos de Teste TDD

## Cadastro de Clientes e Pets



### Cadastro de novo Cliente

```
def test_cadastro_novo_cliente():
    cliente = Cliente(nome="João", email="joao@example.com")
    assert cadastro_cliente(cliente) == "Cadastro realizado com sucesso"
```



### Cadastro de novo Pet

```
def test_cadastro_novo_pet():
    pet = Pet(nome="Rex", tipo="Cachorro", idade=2)
    assert cadastro_pet(pet) == "Pet cadastrado com sucesso"
```

# Casos de Teste TDD

## Agendamento de Serviços

```
...  
Agendamento de Banho  
  
def test_agendamento_banho():  
    agendamento = Agendamento(servico="Banho", data="2024-06-01",  
hora="10:00")  
    assert agendar_servico(agendamento) == "Agendamento confirmado"
```

```
...  
Cancelar Agendamento  
  
def test_cancelamento_agendamento():  
    agendamento_id = 123  
    assert cancelar_agendamento(agendamento_id) == "Agendamento cancelado com  
sucesso"
```

# Casos de Teste TDD

## Gerenciamento de Estoque



Adição de novo Produto

```
def test_adicao_novo_produto():
    produto = Produto(nome="Ração", quantidade=50)
    assert adicionar_produto(produto) == "Produto adicionado com sucesso"
```



Alerta de baixa de Estoque

```
def test_alerta_baixa_estoque():
    produto = Produto(nome="Ração", quantidade=2)
    assert verificar_estoque(produto) == "Alerta: estoque baixo"
```

# Casos de Teste TDD

## Processamento de Pagamentos

```
...  
Pagamento de Serviços  
  
def test_pagamento_servico():  
    pagamento = Pagamento(cliente="João", valor=100.0, metodo="Cartão")  
    assert processar_pagamento(pagamento) == "Pagamento processado com  
    sucesso"
```

```
...  
Relatório de Pagamentos  
  
def test_relatorio_pagamento():  
    relatorio = gerar_relatorio_pagamento()  
    assert relatorio != None
```

# Casos de Teste TDD

## Emissão de Recibos e Notas Fiscais



### Emissão de Recibo

```
def test_emissao_recibo():
    pagamento = Pagamento(cliente="João", valor=100.0, metodo="Cartão")
    assert emitir_recibo(pagamento) == "Recibo emitido com sucesso"
```



### Envio de Recibo por e-mail

```
def test_envio_recibo_email():
    recibo = Recibo(cliente="João", email="joao@example.com")
    assert enviar_recibo_email(recibo) == "Recibo enviado com sucesso"
```



# Obrigado!



ramiroaz@gmail.com



[LinkedIn.com/in/ramiro-arce/](https://www.linkedin.com/in/ramiro-arce/)