

Documentação Detalhada do Sistema "Pra Que Marido?"

Objetivo do Sistema

O sistema "Pra Que Marido?" tem como objetivo conectar clientes a prestadores de serviços em diversas áreas, como encanadores, eletricitas, jardineiros, etc. A plataforma permite que os clientes solicitem serviços e possam avaliar os prestadores de serviços após a conclusão. Além disso, os prestadores podem gerenciar suas solicitações e serviços prestados.

1. Requisitos Funcionais

Cadastro e Autenticação

1. Cadastro de Usuário

- Os usuários podem se cadastrar como **clientes** ou **prestadores de serviços**.
- O cadastro exige informações básicas como nome, email, telefone e senha. Para prestadores, também é necessário especificar a categoria de serviço prestado.

2. Login e Autenticação

- O sistema deve permitir que os usuários façam login utilizando e-mail e senha.
- Autenticação será feita via **Django Auth** ou **JWT** para garantir segurança nas credenciais.

3. Gerenciamento de Perfil

- Usuários podem atualizar suas informações pessoais, incluindo foto de perfil.
-

Solicitação de Serviço

1. Cliente Solicita Serviço

- O cliente pode buscar por prestadores de serviços de acordo com a categoria e localização.
- O cliente pode criar uma solicitação de serviço, incluindo informações sobre o serviço, prazo e uma descrição detalhada.

2. Prestador Aceita/Recusa

- O prestador pode visualizar as solicitações de serviços e aceitar ou recusar.

3. Histórico de Solicitações

- Os clientes e prestadores podem visualizar o histórico de solicitações passadas e futuras.

Avaliação de Serviços

1. Avaliação pelo Cliente

- Após a conclusão do serviço, o cliente pode avaliar o prestador com uma nota de 1 a 5 e deixar um comentário.

2. Visualização das Avaliações

- Prestadores podem visualizar suas avaliações e comentários recebidos.

Administração

1. Painel Administrativo

- O administrador pode visualizar e gerenciar usuários (clientes e prestadores), serviços solicitados, e avaliações feitas pelos clientes.

2. Estrutura de Banco de Dados

Modelos e Relacionamentos

A seguir, está a estrutura detalhada do banco de dados, com as tabelas e os relacionamentos entre elas.

2.1. Tabela usuario

Tabela que armazena as informações dos usuários (clientes e prestadores de serviços).

Campo	Tipo	Descrição
id	Integer (PK)	Identificador único do usuário
nome	Varchar(100)	Nome completo do usuário
email	Varchar(255)	E-mail do usuário (único)
telefone	Varchar(15)	Número de telefone
senha	Char(128)	Senha criptografada
foto	Varchar(255)	Caminho da foto do usuário

Campo	Tipo	Descrição
	5)	
tipo	Char(10)	Tipo do usuário (cliente ou prestador)
data_criacao	DateTime	Data de criação do usuário
data_atualizacao	DateTime	Data de última atualização

Relacionamentos:

- **Relacionamento:** Um usuário pode ser **cliente** ou **prestador**.

2.2. Tabela categoria_servico

Tabela que armazena as categorias de serviços disponíveis (ex: encanador, eletricista).

Campo	Tipo	Descrição
id	Integer (PK)	Identificador único da categoria
nome	Varchar(50)	Nome da categoria (ex: Encanador)
descricao	Text	Descrição da categoria

Relacionamentos:

- **Relacionamento:** Um prestador de serviço pode oferecer vários serviços, cada um associado a uma categoria.

2.3. Tabela servico

Tabela que armazena os serviços oferecidos pelos prestadores.

Campo	Tipo	Descrição
id	Integer (PK)	Identificador único do serviço
prestador_id	Integer (FK)	Chave estrangeira para o usuário (prestador)
categoria_id	Integer (FK)	Chave estrangeira para a categoria
descricao	Text	Descrição detalhada do serviço
preco_estimado	Decimal(10, 2)	Preço estimado do serviço
disponibilidade	Boolean	Se o prestador está disponível

Relacionamentos:

- **Prestador:** Um prestador pode oferecer múltiplos serviços.
- **Categoria:** Um serviço está associado a uma categoria específica.

2.4. Tabela solicitacao_servico

Tabela que armazena as solicitações de serviço feitas pelos clientes.

Campo	Tipo	Descrição
id	Integer (PK)	Identificador único da solicitação
cliente_id	Integer (FK)	Chave estrangeira para o cliente
prestador_id	Integer (FK)	Chave estrangeira para o prestador
servico_id	Integer (FK)	Chave estrangeira para o serviço solicitado
data_solicitacao	DateTime	Data da solicitação
status	Char(20)	Status da solicitação (pendente, concluído, cancelado)
data_conclusao	DateTime	Data de conclusão do serviço (se aplicável)

Relacionamentos:

- **Cliente:** Um cliente pode fazer várias solicitações.
- **Prestador:** Um prestador pode ser atribuído a várias solicitações.
- **Serviço:** Cada solicitação refere-se a um serviço específico.

2.5. Tabela avaliacao

Tabela para armazenar as avaliações feitas pelos clientes sobre os prestadores.

Campo	Tipo	Descrição
id	Integer (PK)	Identificador único da avaliação
cliente_id	Integer (FK)	Chave estrangeira para o cliente
prestador_id	Integer (FK)	Chave estrangeira para o prestador
nota	Integer (1-5)	Nota dada pelo cliente (de 1 a 5)
comentario	Text	Comentário da avaliação
data_avaliacao	DateTime	Data da avaliação

Relacionamentos:

- **Cliente:** Cada avaliação é feita por um cliente.
- **Prestador:** Cada avaliação refere-se a um prestador.

2.6. Relacionamentos entre Tabelas

- **Usuários (cliente e prestador):** Um usuário pode ser tanto um cliente quanto um prestador. Um cliente pode ter várias solicitações de serviço, e um prestador pode ser atribuído a várias solicitações.
 - **Solicitações de Serviço:** Cada solicitação é vinculada a um cliente e a um prestador, e cada solicitação está associada a um serviço específico.
 - **Avaliações:** Cada avaliação é feita por um cliente e é atribuída a um prestador após a execução de um serviço.
-

3. Arquitetura do Sistema

3.1. Arquitetura de Camadas

O sistema será baseado na arquitetura **Model-View-Controller (MVC)**:

1. **Model (Modelo):** Responsável pela representação dos dados no banco de dados.
2. **View (Visão):** Responsável pela interface do usuário (front-end).
3. **Controller (Controlador):** Responsável pela lógica de processamento e comunicação entre as Views e Models.

3.2. Fluxo de Funcionalidade

1. Cadastro de Usuário:

- O usuário se cadastra e escolhe se será cliente ou prestador de serviço.
- Os dados são validados e armazenados no banco de dados.

2. Solicitação de Serviço:

- O cliente realiza uma solicitação, associando um serviço ao prestador.
- O prestador visualiza a solicitação e decide se aceita ou recusa.

3. Avaliação do Serviço:

- Após o serviço ser concluído, o cliente avalia o prestador.
- A avaliação é registrada e associada ao prestador.

3.3. Fluxo de Dados

1. O cliente faz login.
2. O cliente faz uma solicitação de serviço.
3. O prestador aceita ou recusa.
4. O cliente avalia o prestador após o serviço ser concluído.

4. Conclusão

Esta documentação detalha a arquitetura do sistema, as estruturas de banco de dados e os fluxos funcionais que devem ser seguidos para criar o sistema "Pra Que Marido?". Com uma arquitetura modular e fácil de expandir, o sistema pode ser ajustado para incorporar novas funcionalidades, como integração com sistemas de pagamento, chat, ou geolocalização no futuro.