

1. Introdução

1.1 Propósito

Este documento tem como objetivo especificar os requisitos funcionais e não funcionais para o sistema de cadastro de clientes (Pessoa Física e Jurídica) da empresa de venda de equipamentos de energia solar, desenvolvido como parte da disciplina **Programação Web II**.

O sistema possibilita o registro e consulta de usuários com informações pessoais, jurídicas, e de contato.

1.2 Escopo

O sistema permite:

- Cadastro de clientes do tipo **Pessoa Física (PF)** e **Pessoa Jurídica (PJ)**
- Armazenamento de dados pessoais, contato e endereço
- Visualização de lista de clientes cadastrados
- Utilização de **Django** como framework web

Funcionalidades futuras (fora do escopo atual):

- Autenticação para clientes
- Dashboard administrativo completo
- Módulo de vendas e geração de orçamentos
- Integração com API de CEP
- Máscaras dinâmicas para CPF/CNPJ

1.3 Definições e abreviações

Termo	Significado
-------	-------------

PF	Pessoa Física
----	---------------

PJ	Pessoa Jurídica
----	-----------------

CRUD Create, Read, Update,
Delete

Django Framework Web Python
o

2. Descrição Geral

2.1 Perspectiva do Produto

O sistema é uma aplicação web simples de cadastro e consulta, servindo como protótipo funcional para gerenciamento de clientes de uma empresa de energia solar.

2.2 Funcionalidades Principais

- Cadastro de PF
- Cadastro de PJ
- Armazenamento de informações de contato e endereço
- Listagem de clientes cadastrados

2.3 Usuários do Sistema

Usuário	Descrição
Administrador	Gerencia o sistema e consulta os cadastros

2.4 Restrições

- Banco SQLite
 - Tempo de entrega (1 aula — versão MVP)
 - Não possui autenticação do cliente final (somente backend)
-

3. Requisitos Funcionais

Código	Requisito	Descrição
RF01	Cadastro PF	O sistema deve permitir cadastrar um cliente do tipo Pessoa Física
RF02	Cadastro PJ	O sistema deve permitir cadastrar Pessoa Jurídica
RF03	Campos PF	Nome, CPF, RG (opcional), Data Nasc., Email, Telefones, Endereço
RF04	Campos PJ	Razão Social, CNPJ, IE (opcional), Data Abertura, Site (opcional), Email, Telefones, Endereço
RF05	Validação PF/PJ	Campos obrigatórios devem ser verificados para cada tipo
RF06	Listagem	O sistema deve exibir a lista de clientes cadastrados
RF07	Identificar tipo	A listagem deve indicar o tipo do cliente (PF ou PJ)
RF08	Informações na lista	Mostrar nome/razão social, documento, email, telefone, cidade, estado
RF09	Feedback de cadastro	Usuário deve visualizar se o cadastro foi concluído com sucesso
RF10	Uso de rotas Django	Todas funcionalidades devem ser acessadas por rotas web

4. Requisitos Não Funcionais

Código	Requisito	Descrição
RNF01	Linguagem	Python 3 e Django
RNF02	Banco	SQLite padrão Django
RNF03	Interface	HTML/CSS responsivo
RNF04	Usabilidade	Formulário simples e intuitivo
RNF05	Segurança	Uso obrigatório de <code>{% csrf_token %}</code>
RNF06	Portabilidade	Compatível com desktops e mobile
RNF07	Tempo de Resposta	A listagem deve carregar em até 2 segundos

5. Modelo de Dados (Resumo)

Tabelas

- Cliente (extends AbstractUser)
- PessoaFisica (1–1 Cliente)
- PessoaJuridica (1–1 Cliente)

Principais campos:

- Nome/Razão Social
- CPF/CNPJ
- RG/IE
- Data nasc./abertura
- Contatos
- Endereço

6. Casos de Uso (Simplificados)

UC01 — Cadastrar Cliente PF/PJ

Atores: Administrador

Fluxo principal:

1. Acessa formulário
2. Seleciona PF ou PJ
3. Preenche dados
4. Confirma cadastro
5. Sistema salva e retorna mensagem

UC02 — Listar Clientes

Atores: Administrador

Fluxo principal:

1. Acessa página de lista
 2. Sistema exibe todos os clientes cadastrados
-

7. Interface do Usuário

- Página de cadastro com abas PF/PJ
 - Página de listagem com tabela responsiva
-

8. Critérios de Aceitação

- Cliente PF ou PJ cadastrado com sucesso
 - Dados persistidos no banco
 - Página `/usuarios/` exibe registros
 - Testes manuais funcionais concluídos
-

Conclusão

Este documento descreve o conjunto de requisitos do sistema MVP desenvolvido para cadastro e listagem de clientes PF/PJ, atendendo aos objetivos da disciplina e criando base para expansão futura.