

Diagrama DER

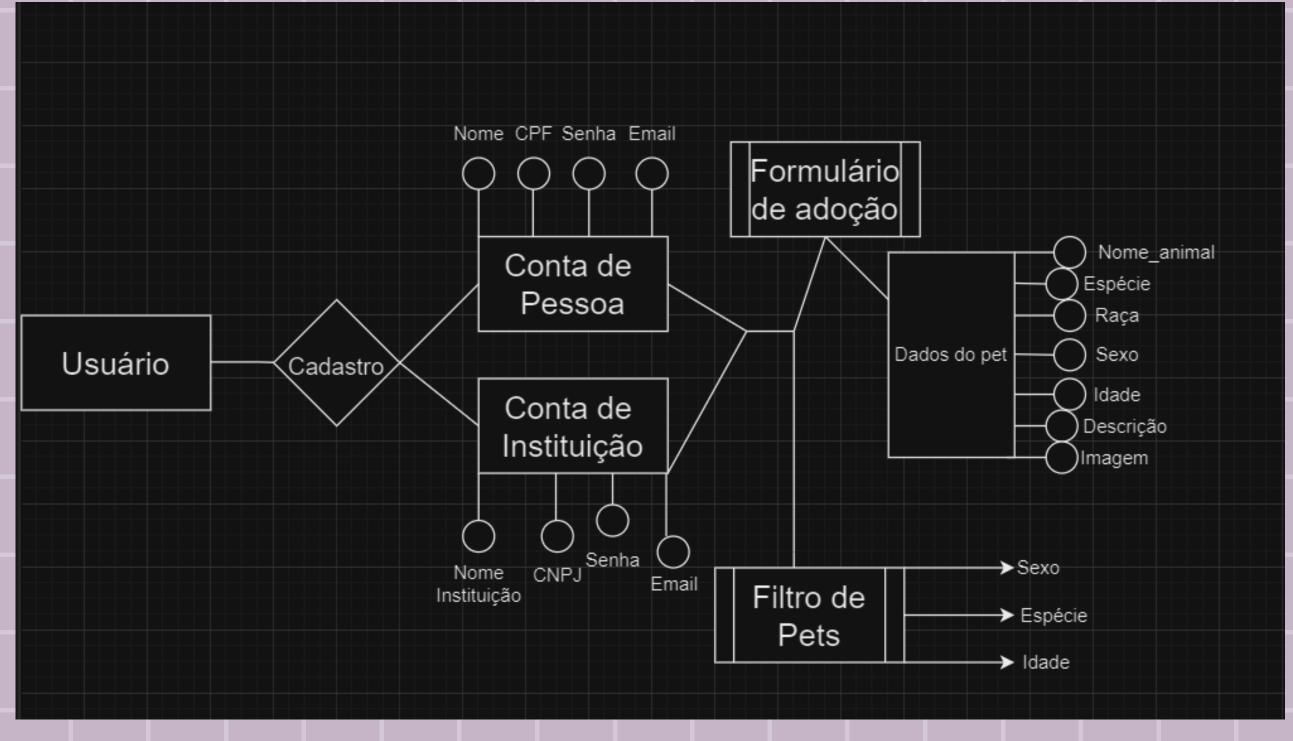








Diagrama DSC

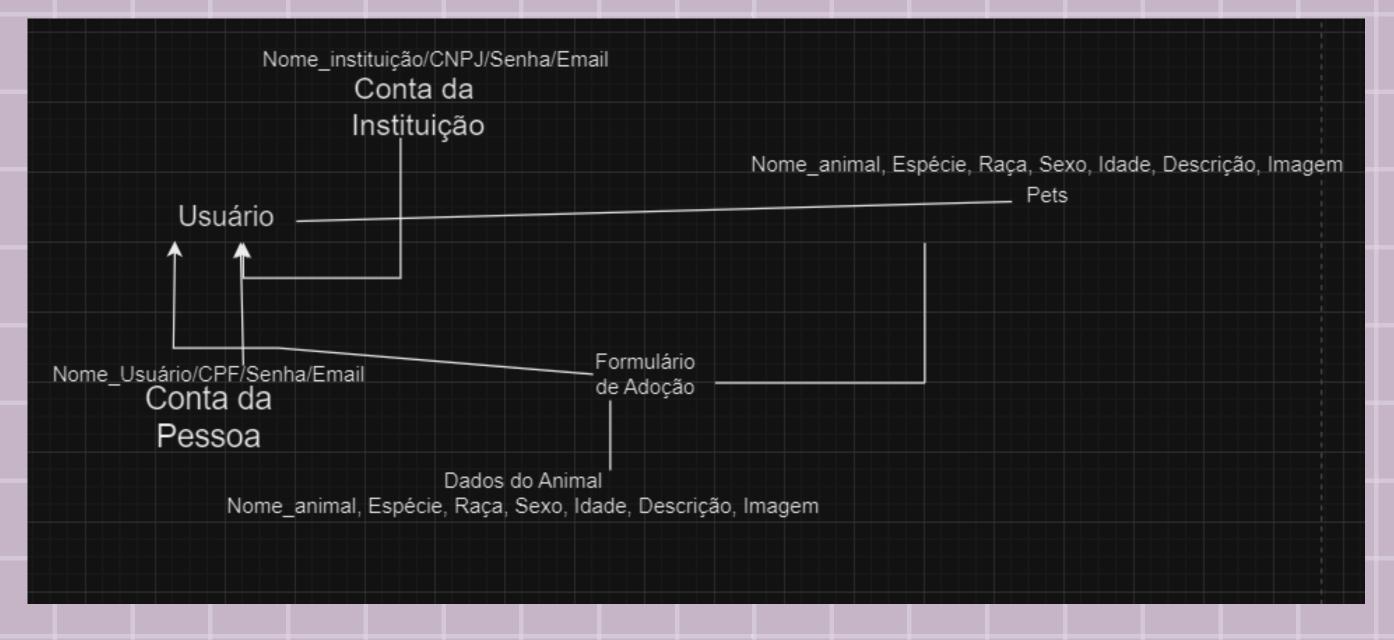
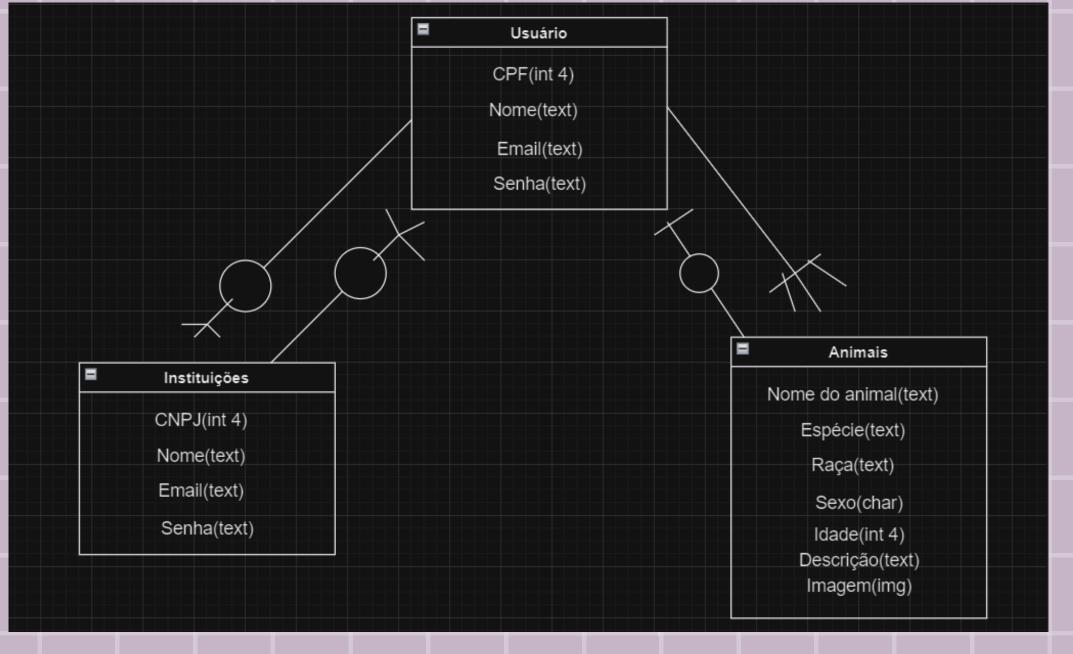








Diagrama DER Extendido









Cadastro de Usuário

Rafael Costa Almeida

12514552260

costaRafael@gmail.com

Criar conta de Usuário

Front-end

```
function salvarDados() {
 // Recupera os valores dos campos de texto
 var nomeInstituicao = document.getElementById("nomeInstituicao").value;
 var cnpj = document.getElementById("cnpjInput").value;
 var senha = document.getElementById("senhaInput").value;
 var contato = document.getElementById("contato").value;
 // Verifica se os campos estão preenchidos
 if (nomeInstituicao && cnpj && senha && contato) {
   // Cria um objeto com os dados
   var dados = {
     nomeInstituicao: nomeInstituicao,
     cnpj: cnpj,
     senha: senha,
     contato: contato
   // Converte o objeto em uma string JSON
   const options = {
     method : 'POST',
     body : JSON.stringify(dados)
   var usuarios = fetch('http://localhost:5554/usuario/insert', options)
```







Back-end



```
//post's e get's para os usuarios
get("/usuario/list", (request, response) -> usuarioService.list(request, response));

post("/usuario/insert", (request, response) -> usuarioService.insert(request, response));

get("/usuario/delete", (request, response) -> usuarioService.delete(request, response));

post("/usuario/update", (request, response) -> usuarioService.update(request, response));

post("/usuario/validate", (request, response) -> usuarioService.validate(request, response));
```

```
public String insert(Request request, Response response) throws Exception{
    UsuarioDAO dao = new UsuarioDAO();

    response.header("Access-Control-Allow-Origin", "*");
    response.header("Access-Control-Allow-Methods", "POST,GET");
    response.header("Access-Control-Allow-Headers", "*");
    response.header("Access-Control-Max-Age", "86400");

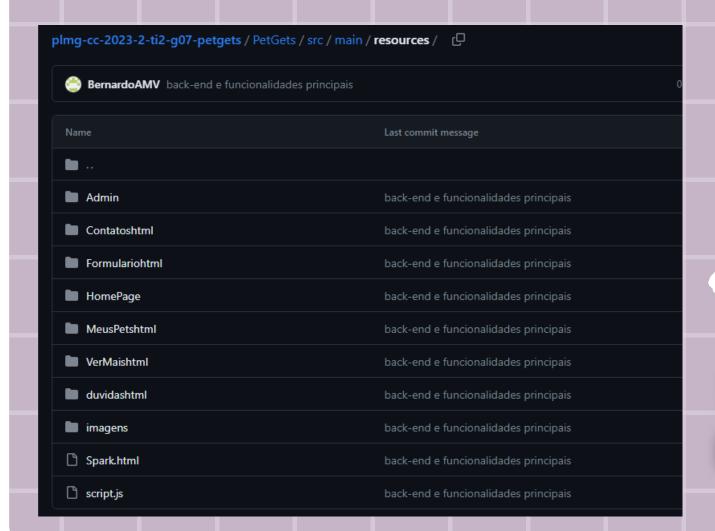
    JSONObject users = new JSONObject(request.body());

    String cpf = users.getString("cnpj");
    String nome = users.getString("nomeInstituicao");
    String senha = users.getString("senha");
    String email = users.getString("contato");
    Usuario user = new Usuario(cpf, nome, senha, email);
    dao.insert(user);

    System.out.println(user.toString());
    return users.toString();
```







SRC/MAIN/ RESOURCES

APP - DAO - MODEL- SERVICE

	plmg-cc-2023-2-ti2-g07-petgets / PetGets / src / main / java / Usuario / 🕒	
	BernardoAMV back-end e funcionalidades principais	
	Name	Last commit message
	арр	back-end e funcionalidades principais
	■ dao	back-end e funcionalidades principais
	model model	back-end e funcionalidades principais
	service service	back-end e funcionalidades principais

IS Canvas

Ferramental de IA

-Um agente que, ao receber certas requisições do usuário, separa palavras-chave, e apresenta pets que correspodem as exigências propostas pelo usuário

-Um algoritmo de agrupamento(Mach ine learning), que agrupa os pets de acordo com os requisites listados pelo usuário

Entradas

- -Primeira mensagem do cliente, iniciando a conversa.
- -Especificação das requisições que o usuário deseja para seu pet.

Saídas

- -Apresentação de pets que se encaixam dentro dos requisites exigidos.
- -Caso o usuário, deseje especificar mais, o sistema continuará "filtrando" animais.
- -Caso não hajam mais animais que se encaixem, ou requisitos inalcançaveis, o sistema informará o usuário disso.

Proposição de valor

- -Dinamizar o atendimento dos usuários.
- -Modernizar e aprimorar a função de filtragem já existente

Equipe

- -Scrum Master
- -Desenvolvedor

Clientes

-Clientes que desejam adotar um pet.

Stakeholders Chaves

- -Investidores.
- -Setor econômico da empresa. -CEO da

empresa

Custos

- -Tempo para a criação e implementação.
- -Custo com equipe para a manutenção do chatbot

Receitas

- -Atendimento mais eficiente e veloz de clientes
- -Site mais dinâmico e atrativo, resultando em mais clientes





