

MY GIFT



Grupo:

Leticia Americano Lucas

Ludmila Bruna Santos Nascimento

Marcos Ani Cury Vinagre Silva

Victor Leite de Andrade

Contexto

- Atualmente sempre que alguém comemora alguma data especial deve-se apresentar essa pessoa, com isso percebemos que tem-se muita dificuldade para escolher o presente dessa pessoa



Projeto

My Gift

Habilidade de presentear

- Contexto da Aplicação: Achar presentes de forma rápida e eficiente
- Objetivo: Ajudar o usuário a achar o presente ideal para alguém

- Objeto: Portal My Gift, Banco de Dados, Pesquisas diversas
- Público-Alvo: Mulheres entre 20 e 45 anos

Funcionalidades do Portal

- Responsividade
- Login e Cadastro
- SAC
- Chatbot para ajuda o usuário
- IA para escolher um presente ideal com base em hobbies

Projeto da interface

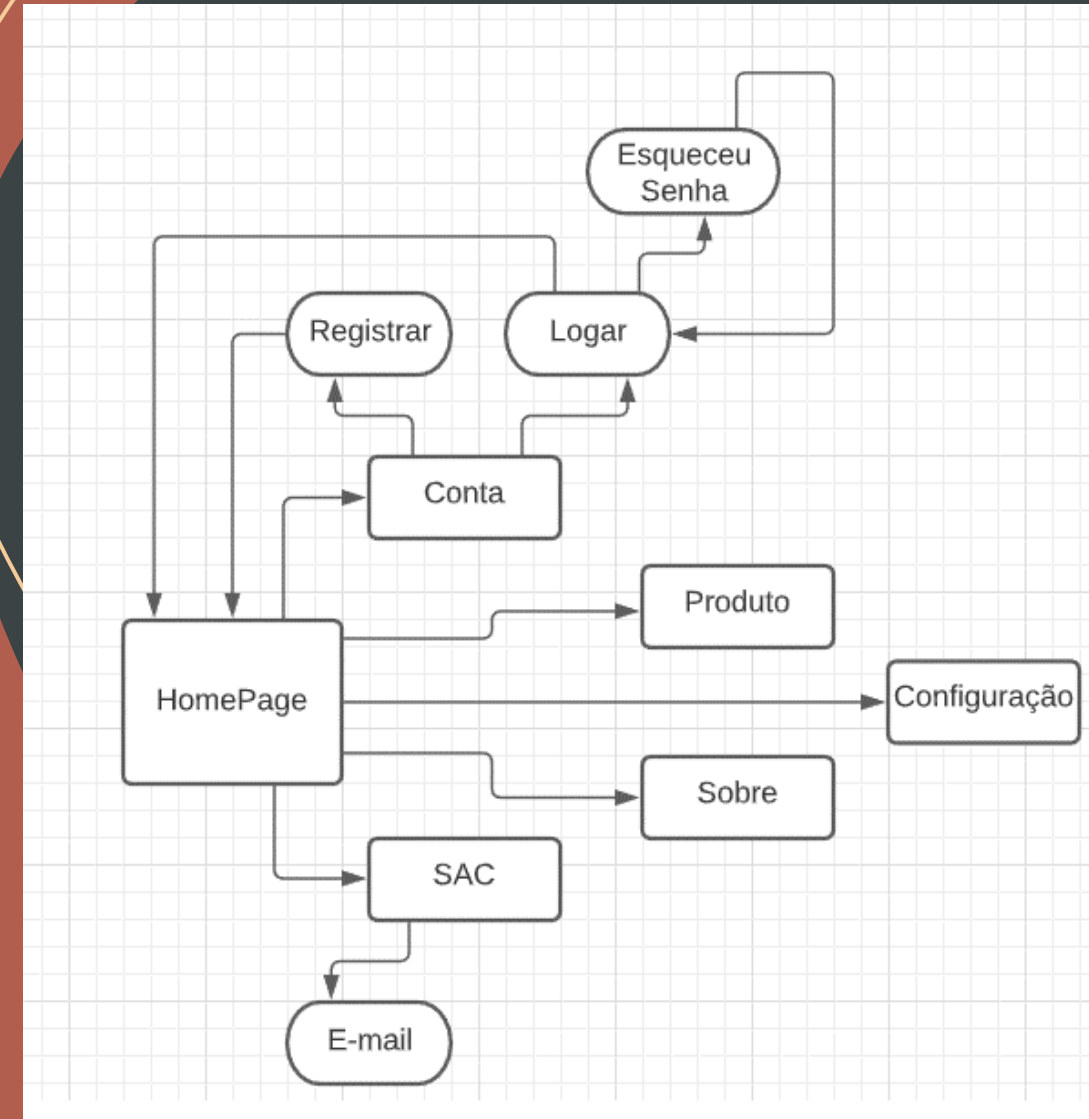
A montagem da interface tem foco:

- Agilidade
- Acessibilidade
- Usabilidade

Assim, a identidade visual é padronizada em todas as telas

Wireframe

- Home Page: Index do site
- Conta: Local para login e cadastro
- Sobre: Breve resumo do portal e seus integrantes
- SAC: Canal aberto para comunicação do cliente com o time



Home

Home Page do portal aonde você tem acesso a todos as outras informações

Cadastro e Login

Cadastro e Login do usuário

Formulário

Formulário para gerar um presente aleatório

SAC

Meio de comunicação do usuário com o grupo

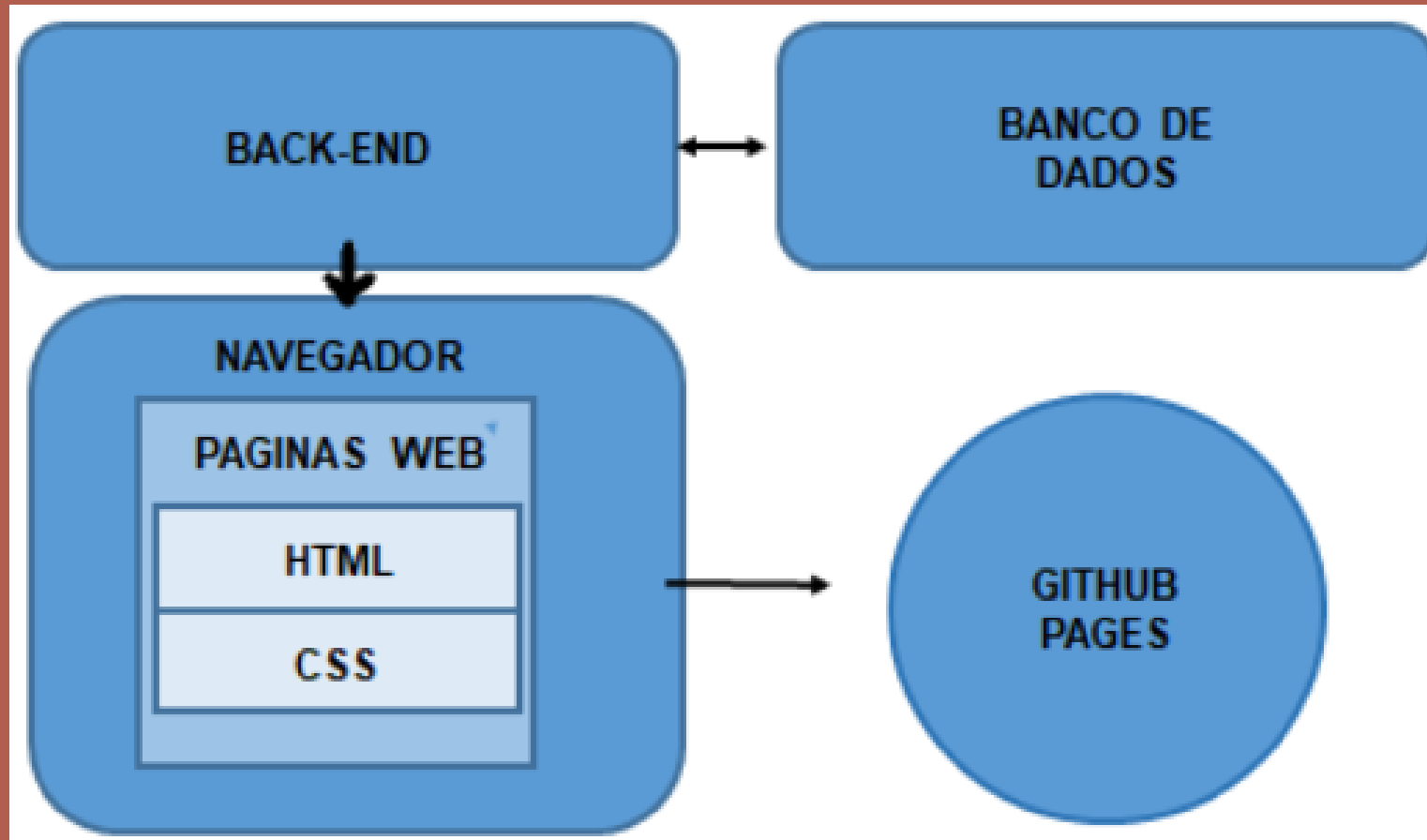
Sobre Nós

Breve resumo do site e dos integrantes do grupo

Produtos

Catálogo de todos os produtos do portal

Programação da Funcionalidades



Classe DAO

```
package produtos;
import java.sql.*;

public class DAO {
    private Connection conexao;

    public DAO() {
        conexao = null;
    }

    public boolean conectar() {
        String driverName = "org.postgresql.Driver";
        String serverName = "localhost";
        String mydatabase = "Produtos";
        int porta = 5432;
        String url = "jdbc:postgresql://" + serverName + ":" + porta + "/" + mydatabase;
        String username = "postgres";
        String password = "*****";
        boolean status = false;

        try {
            Class.forName(driverName);
            conexao = DriverManager.getConnection(url, username, password);
            status = (conexao == null);
            System.out.println("Conexão efetuada com o postgres!");
        } catch (ClassNotFoundException e) {
            System.err.println("Conexão NÃO efetuada com o postgres -- Driver não encontrado -- " + e.getMessage());
        } catch (SQLException e) {
            System.err.println("Conexão NÃO efetuada com o postgres -- " + e.getMessage());
        }

        return status;
    }
}
```

Classe Produto e Principal

```
package produtos;
import produtos.MyIO;
import java.util.Random;
public class FuncaoProdutos {

    public static DAO dao = new DAO();

    public static Produtos Aleatorio()
    {
        Random aleatorio = new Random();
        int valor = aleatorio.nextInt(6)+1;

        Produtos[] produtos = dao.getProdutos();
        int id = valor;
        int id = MyIO.readInt();
        Produtos[] produtos = dao.getProdutos();
        Produtos produto = new Produtos(i, null, i, i);
        boolean isProduto = false;
        do
        {
            if (produtos[i].getId() == id)
            {
                isProduto = true;
                produto = produtos[i];
            }
            i++;
        }while(!isProduto);
        return produto;
    }
    public static void main(String [] args)
    {
        dao.conectar();
        Produtos aleatorio = Aleatorio();
    }
}
```

```
package produtos;

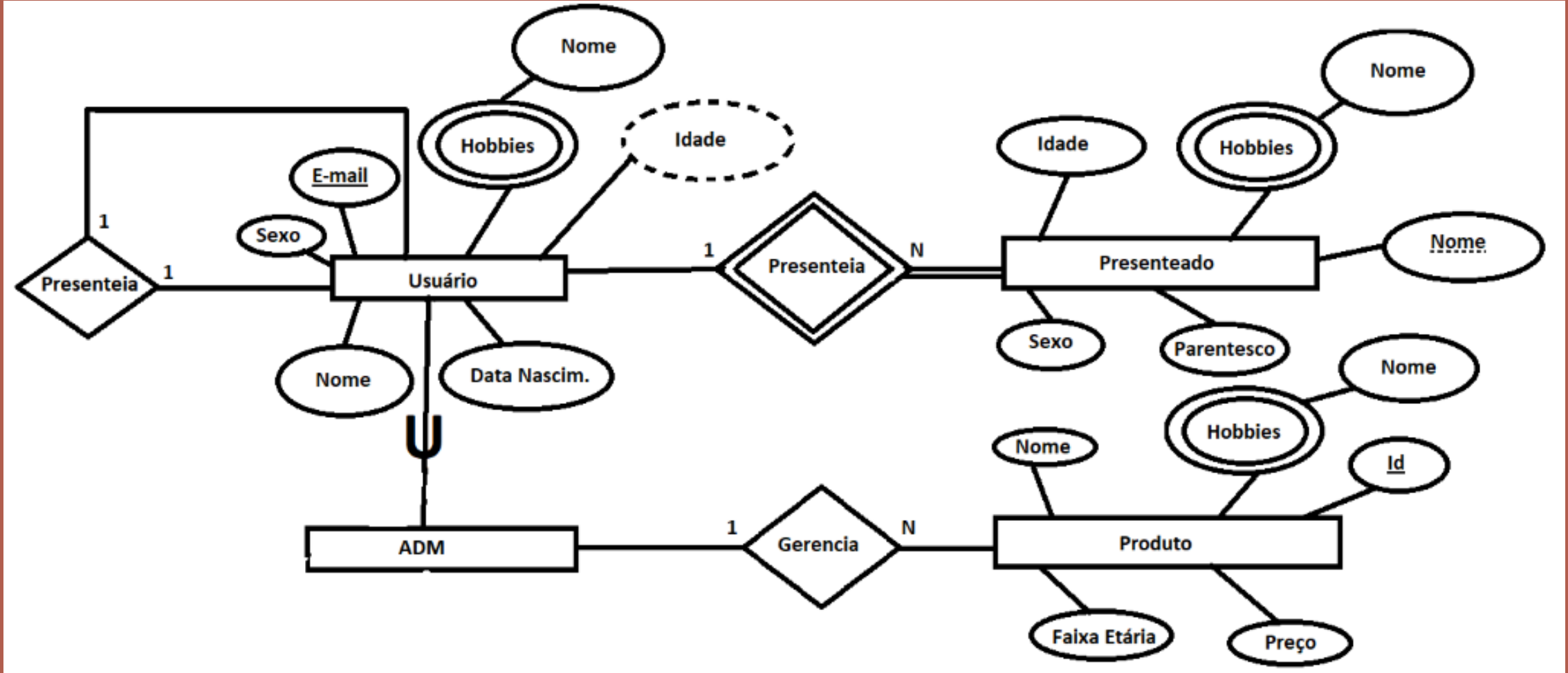
public class Produtos {
    private int id;
    private String nome;
    private double preco;
    private int faixaEtaria;

    /**
     * Criar um novo Produto.
     *
     */

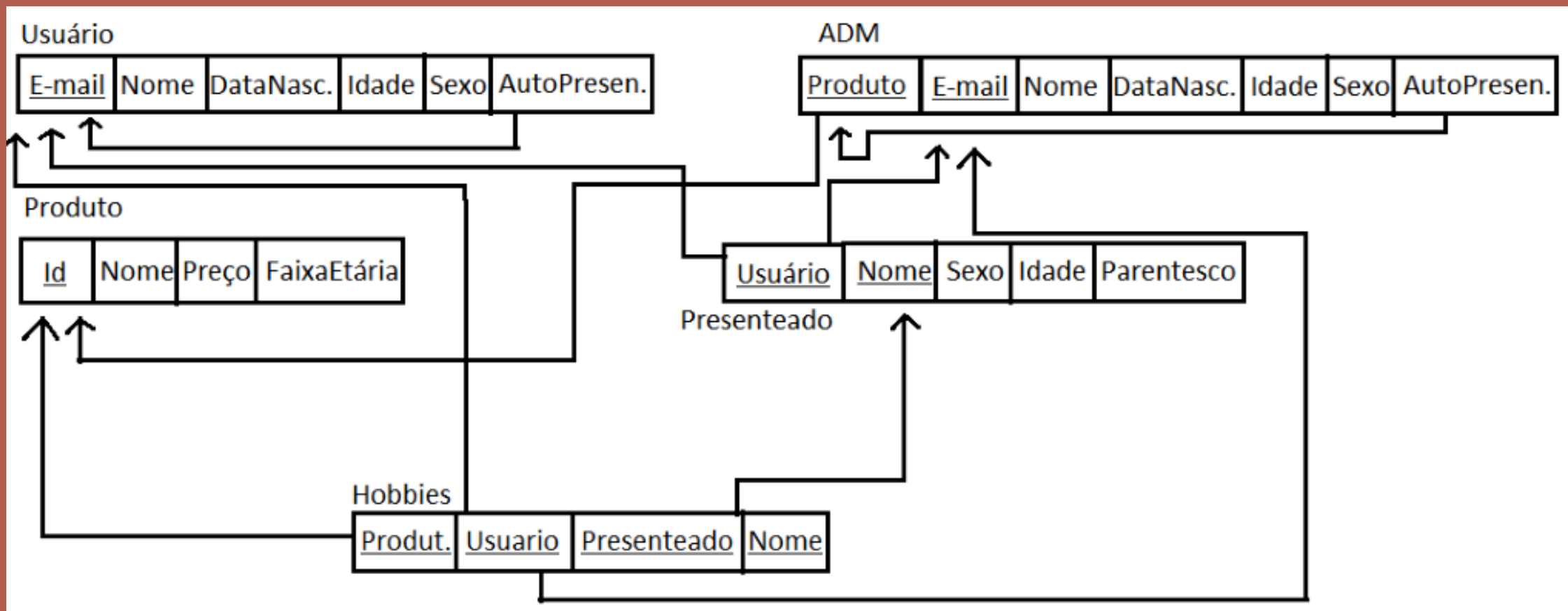
    public Produtos(int id, String nome, double preco, int faixaEtaria) {

        this.id = id;
        this.nome = nome;
        this.preco = preco;
        this.faixaEtaria = faixaEtaria;
    }
}
```

Modelo DER



Modelo Relacional



SQL

phpPgAdmin

PostgreSQL 12.4 (Ubuntu 12.4-0ubuntu0.20.04.1) rodando em localhost:5432 -- Você está logado como usuário "ti2cc" [SQL](#) | [Histórico](#) | [Encontrar](#) | [Sair](#)

phpPgAdmin: PostgreSQL: postgres: MyGift:

[Tabelas](#) [Visões](#) [Seqüências](#) [Funções](#) [Busca por Texto Completo](#) [Domínios](#) [Privilégios](#) [Exportar](#)

Tabela	Proprietário	Tablespace	Número estimado de linhas	Ações										Comentário
<input type="checkbox"/> presenteado	ti2cc		0	Navegar	Selecionar	Inserir	Vazio	Alterar	Deletar	Vácuo	Analisar	Reindexar		
<input type="checkbox"/> produto	ti2cc		0	Navegar	Selecionar	Inserir	Vazio	Alterar	Deletar	Vácuo	Analisar	Reindexar		
<input type="checkbox"/> usuario	ti2cc		0	Navegar	Selecionar	Inserir	Vazio	Alterar	Deletar	Vácuo	Analisar	Reindexar		

Ações sobre múltiplas linhas

Selecionar tudo / Desmarcar tudo ---> [Analisar](#) [Executar](#)

[Criar tabela](#) | [Criar tabela similar](#)

Left Sidebar:

- Servidores
 - PostgreSQL
 - postgres
 - Esquemas
 - MyGift
 - Tabelas**
 - Visões
 - Seqüências
 - Funções
 - Busca por Tex
 - Domínios
 - public