

ANÁLISE E DESENVOLVIMENTO DE SOFTWARE

TÓPICOS INTEGRADORES II

Documentação de Projeto de Software

Versão 1.2

Autor(a): Tarcisio Dantas de Andrade

2020

Controle de Versão do Documento

|  |  |
| --- | --- |
| Versão | Descrição |
| 1.2 | Sistema para gerenciamento de vendas de carros. Modificado Relacionamento entre estoque e venda e estoque e fabricante, adicionado atributo a Admin |

Sumário

[1. Introdução ao Documento 4](#_Toc36326375)

[1.1. Área de negócio do Sistema 4](#_Toc36326376)

[1.2. Principais funcionalidades 4](#_Toc36326377)

[1.3. Método de trabalho 4](#_Toc36326378)

[2. Modelo de Dados 4](#_Toc36326379)

[2.1. Modelo de Visão 4](#_Toc36326380)-5

[2.2. Modelo Conceitual 5](#_Toc36326381)

[2.3. Modelo Lógico 5-7](#_Toc36326382)

[2.4. Dicionário de Dados 7-10](#_Toc36326383)

[2.5. Modelo Físico 10](#_Toc36326384)

[3. Análise e Design 10](#_Toc36326385)

[3.1. Diagrama de Classes 10](#_Toc36326386)

[4. Arquitetura do Software 10](#_Toc36326387)

[4.1. Padrão de projeto 10](#_Toc36326388)

[4.2. Protótipo 10](#_Toc36326389)

1. Introdução ao Documento

## Área de negócio do Sistema

Sistema para gerenciamento de venda de carros.

## Principais funcionalidades

Devera conter um sistema de login para funcionário, com a opção de inserir novos usuários, consulta no estoque, consulta do cliente e venda de carro.

## Método de trabalho

O método de desenvolvimento ágil com a linguagem Java usando a biblioteca Swing, o SGBD a ser utilizado é o MySQL.

1. Modelo de Dados

## Modelo de Visão

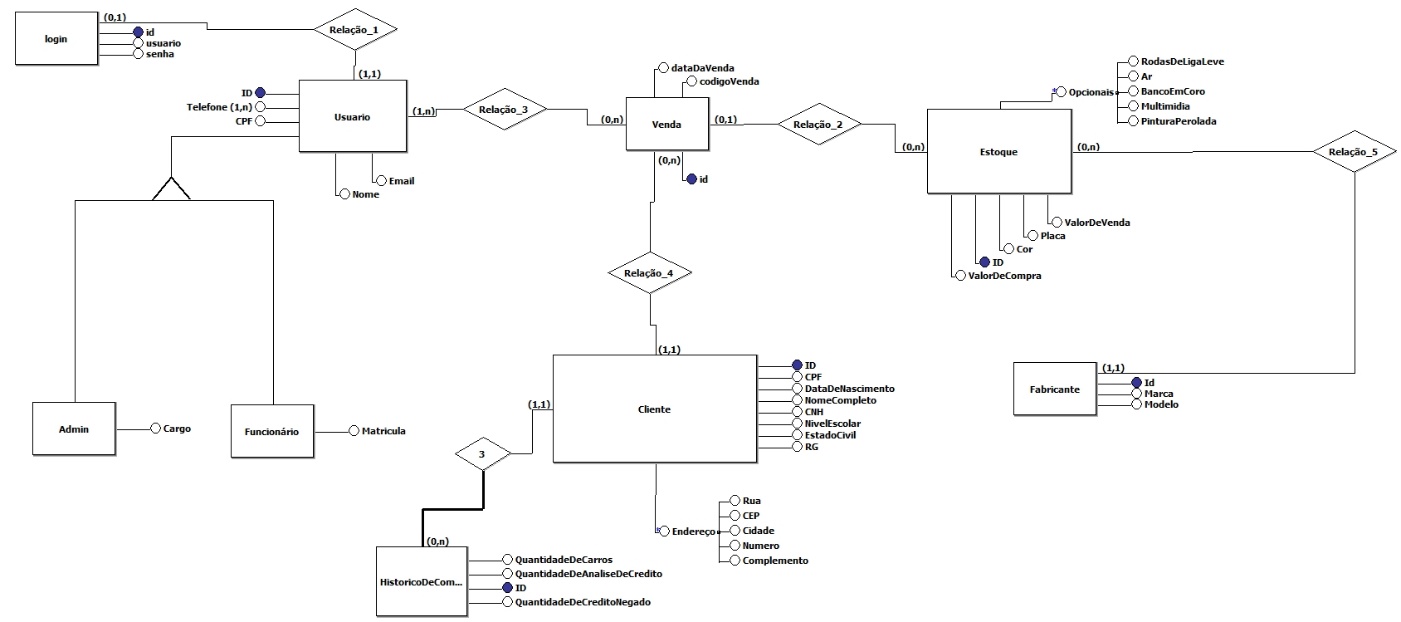
Loja de Carros

Na loja de carros do senhor Osvaldo deve conter uma área de login para os funcionários poderem acessar, somente funcionários cadastrados pelo admin podem se conectar ao sistema, dentro do sistema deve conter como cadastrar novos funcionários para acessar o sistema, deve ter como adicionar novos produtos, como consultar novos produtos, cadastrar um cliente e ligar um cliente a venda especifica, ligar o cliente a uma reserva, se o carro for “0 km” deve conter a opção do cliente escolher os opcionais disponível para o carro. Depois da compra o veiculo deve ser subtraído do banco de dados, porém deve constar como venda e o valor da venda deve ficar no sistema. Podendo saber quais carros o usuário já comprou, se teve analise de créditos negadas, e quantas vezes ele fez a analise de crédito, e contendo todos os itens organizados em estoque. Deve conter o login do usuário, no caso armazenar nome e senha do usuário, na tabela usuário deve ser armazenado o os telefones ,CPF, Nome e Email podendo o usuário ser admin ou funcionário normal onde este deve ser munido de matrícula, a venda deve conter a data da venda e o código da venda, já na tabela cliente deve conter o cpf, data de nascimento, nome completo, cnh, nível escolar, estado civil, rg, com o endereço completo contendo rua, cep ,cidade, numero e complemento, deve conter ainda um historio de compras para o cliente contendo quantidade de carros, quantidade de analise de crédito e quantidade de credito negado, já no estoque deve conter valor de venda placa, cor, valor de compra, e os opcionais que são rodas de liga leve, ar, banco em couro multimidia e pintura perolada, deve conter a opção de cadastrar o fabricante com a marca e modelo dos carros dos fabricantes específicos.

***Demais alterações deve ser feita durando a necessidade do projeto sendo obrigatoriamente modificados no modelo de visão e na documentação do projeto.***

## Modelo Conceitual

Modelo conceitual do projeto de banco de dados utilizando a modelagem ER e apresentando o DER.



## Modelo Lógico

Modelo lógico do projeto na 3ª forma normal. Apresentado de forma descritiva. Com todas as constraint de primary key e foreign key.

Segue abaixo as tabelas

=> login(usuario,senha, codUser)

usuario primary key

codUser foreign key referencia tabela usuario

=> venda(dataDaVenda, codigoVenda, codCliente)

codigoVenda primary key

codCliente foreign key refenrencia a tabela cliente

=> Cliente(ID, CPF, DataDeNascimento,NomeCompleto,CNH, NivelEscolar, EstadoCivil, RG, Rua, CEP, Cidade, Numero, Complemento)

ID primary Key

=> Estoque (ID, ValorDeCompra, Cor, Placa, ValorDeVenda, RodasDeLigaLeve, Ar, BancoEmCoro, Multimidia, PinturaPerolada, idVenda,idFabricante)

ID primary Key

idVenda foreign key referencia a tabela venda

idFabricante foreign key referencia a tabela fabricante

=> Fabricante (id, nomeDoFabricante, modelo, marca)

Id primary key

=> telefone(idUsuario, idTelefone)

idUsuario, idTelefone primary key

idUsuario foreign key referencia Usuario

=> HistoricoDeCompras(ID, QuantidadeDeCarros, QuantidadeDeAnaliseDeCredito, QuantidadeDeCreditoNegado, codCliente)

ID, codCliente primary key

=> usuario(ID,CPF, Email, Nome, Matricula, cargo )

ID primary Key

=> Relação\_3 (codVenda, codUsuario)

codVenda, codUsuario primary key

codVenda, foreign key referencia a tabela Venda

codUsuario ,foreign key referencia a tabela Usuario

## Dicionário de Dados

Segue abaixo:

Tabela Login

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| Id | int | Sim | PK | Chave primaria da tabela |
| Usuario | varchar | Sim | UN | Unique |
| senha | varchar | sim |  | Senha de login |
| codUsuario | int | sim | FK | Referencia usuario (ID) |

Tabela venda

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| codigoVenda | int | Sim | PK | Chave primaria da tabela |
| dataDaVenda | date | sim |  | Data da Venda |
| codCliente | int | sim | FK | Referencia cliente (ID) |

Tabela Cliente

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| ID | int | Sim | PK | Chave primaria da tabela |
| CPF | varchar | sim | UN | CPF do Cliente UNIQUE |
| DataDeNascimento | date | sim |  | Data de Nascimento do Cliente |
| NomeCompleto | varchar | Sim |  | Nome completo do Cliente |
| CNH | Varchar | Sim | UN | CNH do Cliente UNIQUE |
| NivelEscolar | Varchar | Não |  | Nível escolar do Cliente |
| EstadoCivil | Varchar | Sim |  | Estado Civil do Cliente |
| RG | Varchar | Sim | UN | RG do Cliente UNIQUE |
| Rua | Varchar | Sim |  | Rua |
| CEP | Varchar | Sim |  | CEP |
| Cidade | Varchar | Sim |  | Cidade |
| Numero | Int | Sim |  | Número da Residência |
| Complemento | varchar | Não |  | Complemento do Endereço |

Tabela Estoque

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| ID | int | Sim | PK | Chave primaria da tabela |
| ValorDeCompra | float | sim |  | Valor que foi comprado da fabrica |
| Cor | varchar | sim |  | Cor do veiculo |
| Placa | varchar | Sim | UN | Placa do Veiculo se tiver UNIQUE |
| ValorDeVenda | float | Sim |  | Valor que foi vendido para o cliente |
| RodasDeLigaLeve | Varchar | Não |  | Se o cliente selecionar como adicional as Rodas de Liga L |
| Ar | Varchar | Não |  | Se o cliente selecionar como adicional o Ar Condicionado |
| BancoEmCoro | Varchar | Não |  | Se o cliente selecionar como adicional Banco em Coro |
| Multimidia | Varchar | Não |  | Se o cliente selecionar como adicional a Multimidia |
| PinturaPerolada | Varchar | Não |  | Se o cliente selecionar como adicional a Pintura Perolada |
| codVendas | Int | Não | FK | Referencia Venda(codigoVenda) |
| codFabricante | Int | Não | FK | Referencia Fabricante(id) |

Tabela Fabricante

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| Id | int | Sim | PK | Chave primaria da tabela |
| modelo | Varchar | sim |  | Modelo do Veiculo |
| Marca | Varchar | Sim |  | Marca do Veiculo |

Tabela telefone

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| idUsuario | int | Sim | PK | Chave primaria da tabela |
| idTelefone | Int | sim | PK | Chave Primaria da tabela |
| idUsuario | int | sim | FK | Referencia usuario (ID) |

Tabela HistoricoDeCompras

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| ID | int | Sim | PK | Chave primaria da tabela |
| codCliente | Int | sim | PK | Chave Primaria da tabela |
| codCliente | Int | sim | FK | Referencia cliente(id) |
| QuantidadeDeCarros | int | sim |  | Quantidade de Carros que o usuario comprou |
| QuantidadeDeAnaliseDeCredito | Int | Sim |  | Quantidade de analise de credito que o usuario fez |
| QuantidadeDeCreditoNegado | Int | Sim |  | Quantidade de vezes que foi negado crédito |

Tabela usuario

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| ID | int | Sim | PK | Chave primaria da tabela |
| CPF | Varchar | sim | UN | CPF do Usuario UNIQUE |
| Email | Varchar | sim | UN | Email do Usuario UNIQUE |
| Nome | Varchar | Sim |  | Nome do Usuario |
| Matricula | Int | Sim | UN | Matricula do Usuario UNIQUE |
| Cargo | Varchar | Não |  | Cadastrar de Gerente para Cima (quem tem cargo não tem matricula) |

Tabela Relação\_3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador | Tipo (tamanho) | Obrigatório | Chave | Descrição |
| codVenda | int | Sim | PK | Chave primaria da tabela |
| codUsuario | Int | sim | PK | Chave Primaria da tabela |
| codVenda | int | sim | FK | Referencia Venda (ID) |
| codUsuario | Int | Sim | FK | Referencia Usuario (Id) |

## Modelo Físico

Apresentando os scripts DDL do banco de dados.

**use** sigcar;

**drop** **database** sigcar;

**create** **database** sigcar **character** **set** UTF8 **collate** utf8\_general\_ci;

**use** sigcar;

**create** **table** usuario(

id **int** **primary** **key** **auto\_increment**,

cpf **varchar**(30) **not** **null** **unique**,

email **varchar**(60) **not** **null** **unique**,

nome **varchar**(30) **not** **null**,

matricula **int** **not** **null** **unique**,

cargo **varchar**(30)

);

**create** **table** login(

id **int** **primary** **key** **auto\_increment**,

usuario **varchar**(60) **not** **null** **unique**,

senha **varchar**(32) **not** **null**,

codUsuario **int** **not** **null**,

**foreign** **key** (codUsuario) **references** usuario (id)

);

**create** **table** telefone(

idTelefone **int** **not** **null**,

idUsuario **int** **not** **null**,

**primary** **key**(idUsuario, idTelefone),

**foreign** **key**(idUsuario) **references** sigcar.usuario (id)

);

**create** **table** fabricante(

id **int** **primary** **key** **auto\_increment**,

marca **varchar**(20) **not** **null**,

modelo **varchar**(20) **not** **null**

);

**create** **table** cliente(

id **int** **primary** **key** **auto\_increment**,

cpf **varchar**(50) **not** **null** **unique**,

dataDeNascimento **date** **not** **null**,

nomeCompleto **varchar**(100) **not** **null**,

cnh **varchar**(50) **not** **null**,

nivelEscolar **varchar**(50),

estadoCivil **varchar**(50) **not** **null**,

rg **varchar**(50) **not** **null** **unique**,

rua **varchar**(50) **not** **null**,

cep **varchar**(50) **not** **null**,

cidade **varchar**(50) **not** **null**,

numero **int** **not** **null**,

complemento **varchar**(60)

);

**create** **table** historicoDeCompras(

id **int** **auto\_increment**,

codCliente **int**,

quantidadeDeCarros **int** **not** **null**,

quantidadeDeAnaliseDeCredito **int** **not** **null**,

quantidadeDeCreditoNegado **int** **not** **null**,

**primary** **key** (id, codCliente),

**foreign** **key**(codCliente) **references** cliente(id)

);

**create** **table** venda (

codigoVenda **int** **auto\_increment**,

dataDaVenda **date** **not** **null**,

codCliente **int**,

**primary** **key**(codigoVenda, codCliente),

**foreign** **key** (codCliente) **references** cliente (id)

);

**create** **table** estoque(

id **int** **primary** **key** **auto\_increment**,

valorDeCompra **float** **not** **null**,

cor **varchar**(20) **not** **null**,

placa **varchar**(20) **unique**,

valorDeVenda **float** **not** **null**,

rodasDeLogaLeve **varchar**(20),

bancoEmCoro **varchar**(20),

multimidia **varchar**(20),

pinturaPerolada **varchar**(20),

codVendas **int**,

codFabricante **int**,

**foreign** **key**(codVendas) **references** venda(codigoVenda),

**foreign** **key**(codFabricante) **references** fabricante(id)

);

**create** **table** relacao\_3(

codVenda **int**,

codUsuario **int**,

**primary** **key** (codUsuario, codVenda),

**foreign** **key**(codUsuario) **references** usuario(id),

**foreign** **key**(codVenda) **references** venda (codigoVenda)

);

1. Análise e Design

## Diagrama de Classes

Será apresentado o diagrama de classes básicas e deverá ser informada qual ferramenta foi utilizada e apresentada a imagem final do diagrama.

1. Arquitetura do Software

## Padrão de projeto

Deverá ser apresentada uma breve descrição sobre o padrão MVC e descrever as classes que serão implementadas em cada uma das camadas.

## Protótipo

Deverão ser apresentadas cada uma das telas do sistema para cada uma das camadas.

Para cada tela deverão ser descritos os requisitos funcionais que a mesma atende.