

SISTEMA DE ENSINO PRESENCIAL CONECTADO

Tecnologia em Análise e Desenvolvimento de Sistemas

ALEXANDRE BRAGA

ARLEI FABIANO HAEFLIGER

DOUGLAS DE PAULA GONÇALVES

LUIS CARLOS DA SILVA

NEYMAR DANIEL CORRÊA

ROBERTO INACIO MOLON

**PRODUÇÃO TEXTUAL EM GRUPO**

Desenvolvimento de Sistemas Controle de Frotas

Caxias do Sul

2015

ALEXANDRE BRAGA

ARLEI FABIANO HAEFLIGER

DOUGLAS DE PAULA GONÇALVES

LUIS CARLOS DA SILVA

NEYMAR DANIEL CORRÊA

ROBERTO INACIO MOLON

**PRODUÇÃO TEXTUAL EM GRUPO**

Desenvolvimento de Sistema Controle de Frotas

Trabalho apresentado ao Curso Superior de Tecnologia em Análise e Desenvolvimeno de Sistema da Universidade Norte do Paraná - UNOPAR, para as disciplinas de:

Análise Orientada a Objetos II

Banco de Dados II

Programação Orientada a Objetos

Programação Web I

Seminário IV.

Orientadores:

Prof. Adriana Ap. Loper

Disciplina: Seminário IV

Prof. Luiz Claudio Perini

Disciplina: Analise Orientada a Objetos II

Prof. Márcio Roberto Chiaveli

Disciplina: Programação Orientada a Objetos II

Prof. Roberto Y. Nishimura

Disciplina: Banco de Dados II

Prof. Veronice de Freitas

Disciplina: Programação WEB I

Caxias do Sul

2015

SUMÁRIO

[2. OBJETIVO 5](#_Toc434442169)

[3. DESENVOLVIMENTO 6](#_Toc434442170)

[3.1. ESTUDO DE CASO. 6](#_Toc434442171)

[3.2. ANÁLISE ORIENTADA A OBJETOS. 7](#_Toc434442172)

[3.2.1. Diagrama de Caso de Uso: 7](#_Toc434442173)

[3.2.2. Diagrama de Classes: 8](#_Toc434442174)

[3.2.3. Diagrama de Máquina de Estados: 8](#_Toc434442175)

[3.2.4. Diagrama de Sequência: 10](#_Toc434442176)

[3.2.5. Diagrama de Implantação: 12](#_Toc434442177)

[3.3. BANCO DE DADOS. 12](#_Toc434442178)

[3.3.1. Modelo Relacional Normalizado: 13](#_Toc434442179)

[3.3.2. Modelo Entidade Relacionamento: 14](#_Toc434442180)

[3.3.3. Diagrama Entidade Relacionamento: 14](#_Toc434442181)

[3.4. PROGRAMAÇÃO ORIENTADA A OBJETOS. 15](#_Toc434442182)

[3.4.1. Geração de Modelo físico. 16](#_Toc434442183)

[3.5. PROGRAMAÇÃO WEB. 27](#_Toc434442184)

[3.5.1. Hypertext Preprocessor o PHP: 27](#_Toc434442185)

[3.5.2. Geração de códigos Web. 30](#_Toc434442186)

[3.5.3. Geração do Formulários. 37](#_Toc434442187)

[3.5.4. Geração das Databases. 38](#_Toc434442188)

[4. CONCLUSÃO 44](#_Toc434442189)

[5. REFERÊNCIA 45](#_Toc434442190)

**1. INTRODUÇÃO**

Estamos desenvolvendo este trabalho com base em Programação C# e PHP não podemos esquecer das Modelagens do projeto e da Programação Orientada a Objetos.

Na Análise Orientada a Objetos, onde é desenvolvido o sistema utilizamos o conceito de objetos onde ele interage entre si.

No Banco de Dados onde são armazenados os dados faremos os Diagramas do projeto.

Com a Programação Orientada a Objeto, devemos adotar uma linguagem de programação onde a forma de aprender se da de maneira objetiva para realizar a programação neste caso usaremos o C#, com a tarefa de fazermos o cadastro.

E com a programação WEB, o desenvolvimento de parte do sistema será feito em PHP.

Quem está querendo fazer uso e lucrar com este mercado de programação ou desenvolver algum sistema precisa compreender bem todos os procedimentos que envolve essa área de trabalho porque vivemos num mercado muito competitivo e com excelentes profissionais, não precisamos ser mais um neste mercado, precisamos fazer a diferença.

#### 2. OBJETIVO

O Objetivo do trabalho é demonstrar o uso da Análise Orientada a Objeto é apresentar diagramas UML para o desenvolvimento do sistema.

Traremos e faremos os diagramas do sistema para ser usado com o banco de dados MySQL, com a Programação orientada a Objetos Usaremos a IDE do C# e desenvolveremos parte de sistema em PHP na programação WEB.

Existira no trabalho também fragmentos dos códigos de programação C#, teremos partes do desenvolvimento em PHP e também mostraremos algumas estruturas para o Banco de Dados.

#### 3. DESENVOLVIMENTO

Com o Inicio da Atividades em Analise e Desenvolvimento de Sistemas, temos o conhecimento suficiente para apresentar neste trabalho partes de um programa para usuário final.

Teremos os diagramas para uso no Banco de Dados MySQL, desenvolveremos os Diagramas necessários para ser implementados neste projeto que visa controlar uma frota de carros.

Usaremos IDE C# para fazer o cadastro de funcionário e veículos e finalizaremos no desenvolvimento com PHP o controle das frotas e apresentação de formulários.

#### 3.1. ESTUDO DE CASO.

Neste trabalho vamos trabalhar com a seguinte Orientação para desenvolver nosso sistema, portando no ciclo do desenvolvimento da análise e programação, temos alguns passos ordenados para seguidos no desenvolvimento do sistema de Controle de Veículos que são:

Nessa empresa teremos 40 funcionários, os veículos terão como características, fabricante, modelo, cor, placa, tipo de combustível e portas, cada funcionário com autorização poderá usar o veículo, será exigido o preenchimento do formulário de solicitação de reserva e encaminhar o setor de frotas preenchendo a identificação do funcionário, data e hora, quantidade de passageiro e destino.

No setor de frotas será verificado a disponibilidade para realizar ou não a reserva caso sim alocar ao formulário e devolvendo ao solicitante e caso não retorna à indisponibilidade e encerrando a solicitação e por fim se confirma a data e hora para retirar o veículo anotando a quilometragem quando retornar o veículo preencher o formulário com data e hora da quilometragem final.

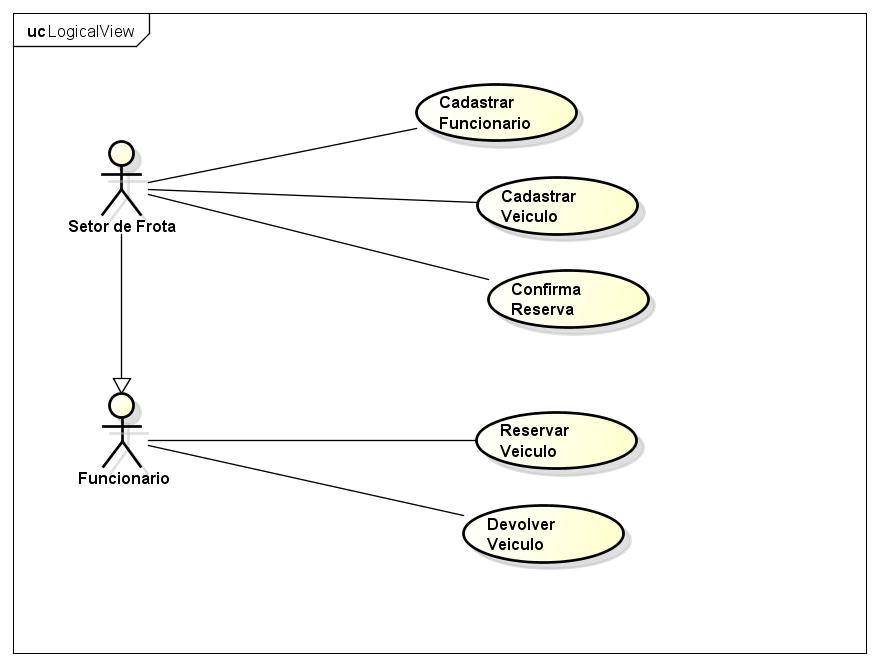
#### 3.2. ANÁLISE ORIENTADA A OBJETOS.

A Análise Orientada a Objetos se inicia criação de um modelo descritivo e possui informações do projeto com detalhes do problema a ser solucionado ou sistema a ser implementado.

Abaixo segue imagens dos Digramas usados em Nosso projeto de Locação de Veículos:

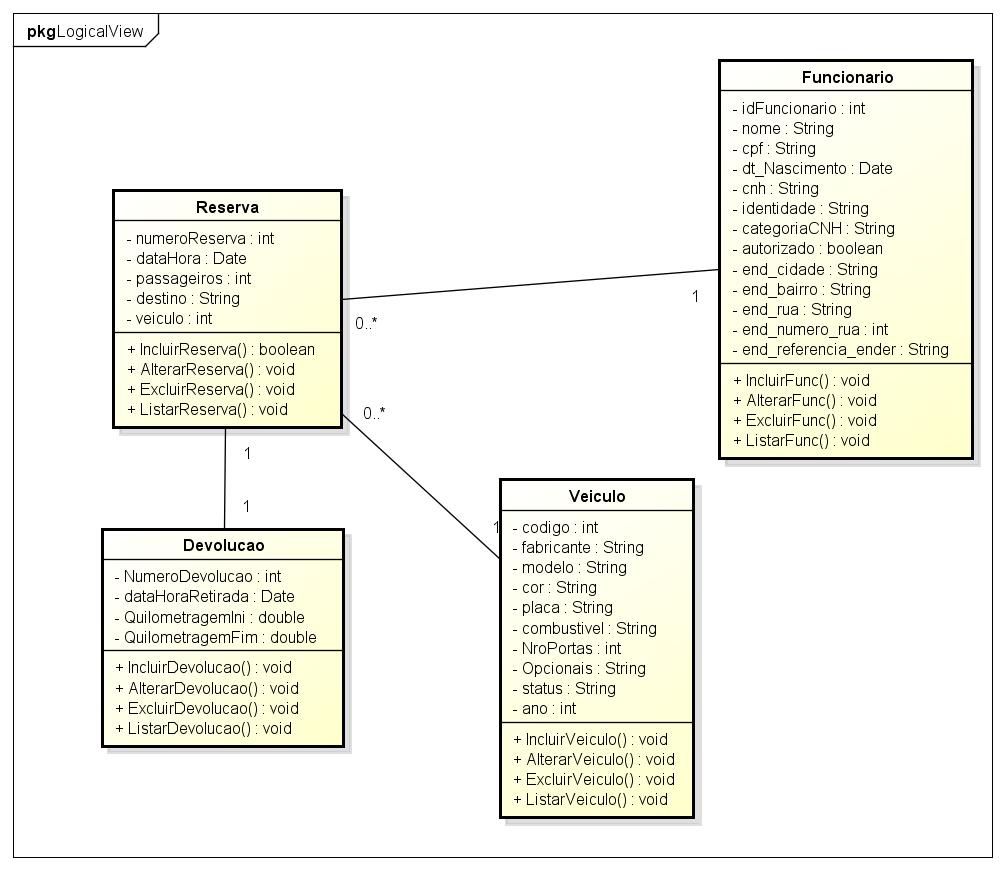
#### 3.2.1. Diagrama de Caso de Uso:

Os Diagramas de Caso de Uso especificam o comportamento do sistema em partes ou conjunto de cenários.



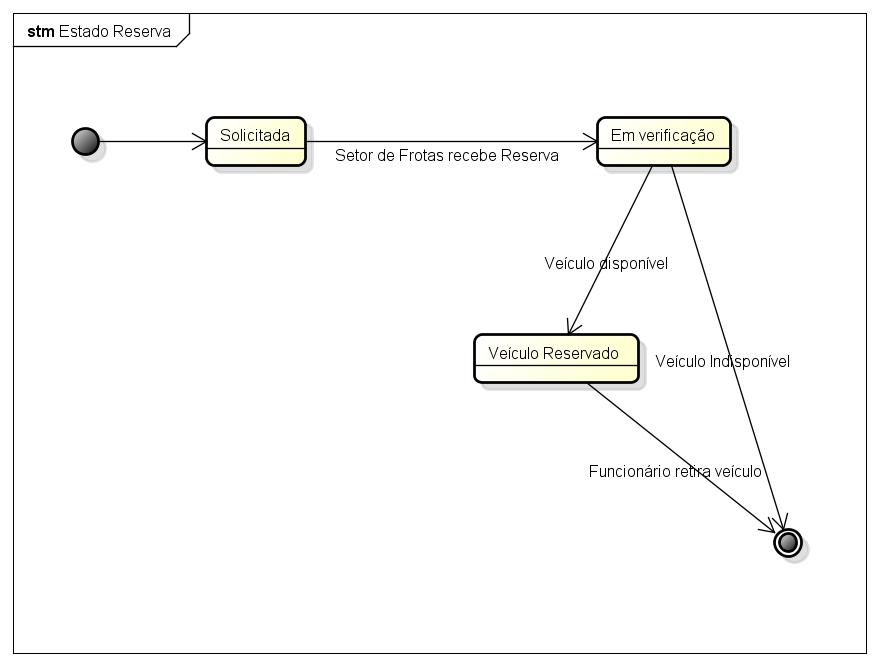
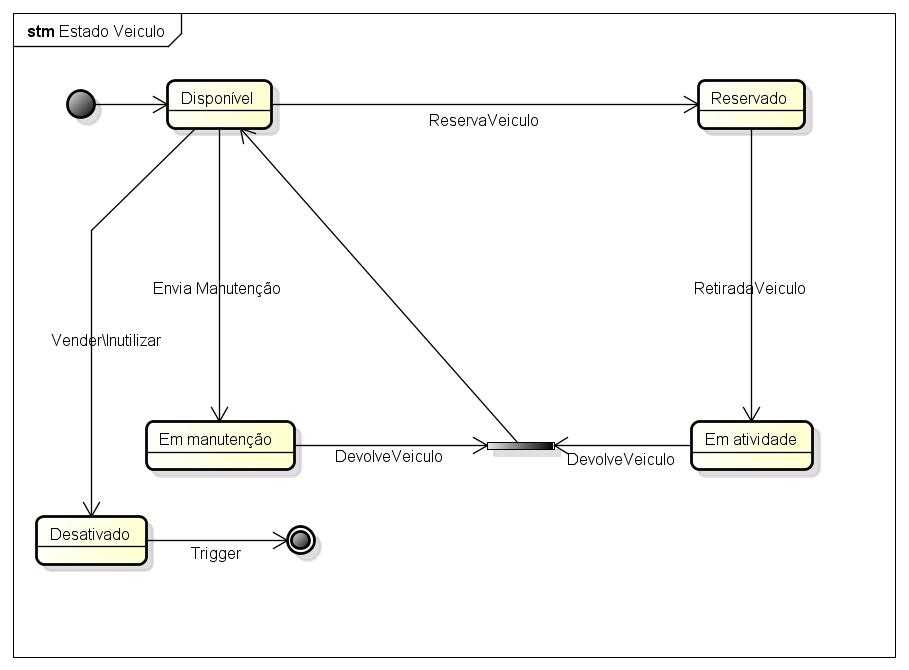
#### 3.2.2. Diagrama de Classes:

Neste diagrama temos o detalhamento das entidades envolvidas através de uma representação estrutural com as classes que servem de modelo para os objetos.



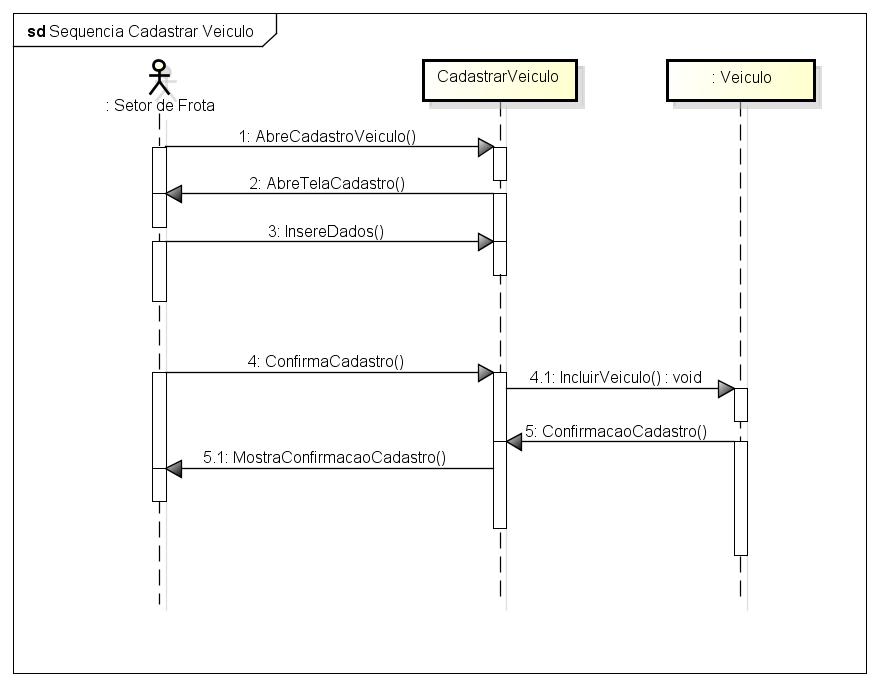
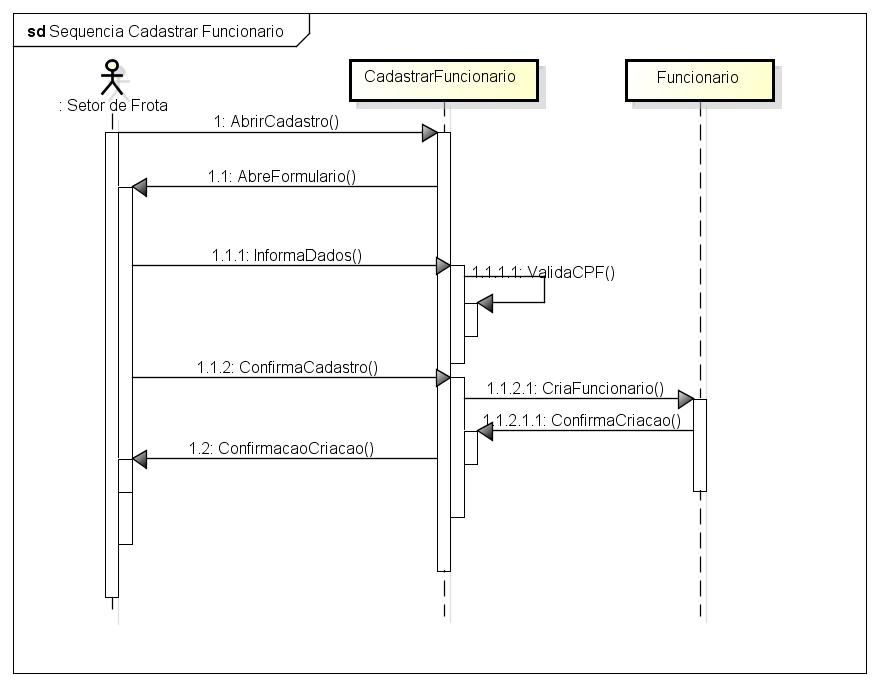
#### 3.2.3. Diagrama de Máquina de Estados:

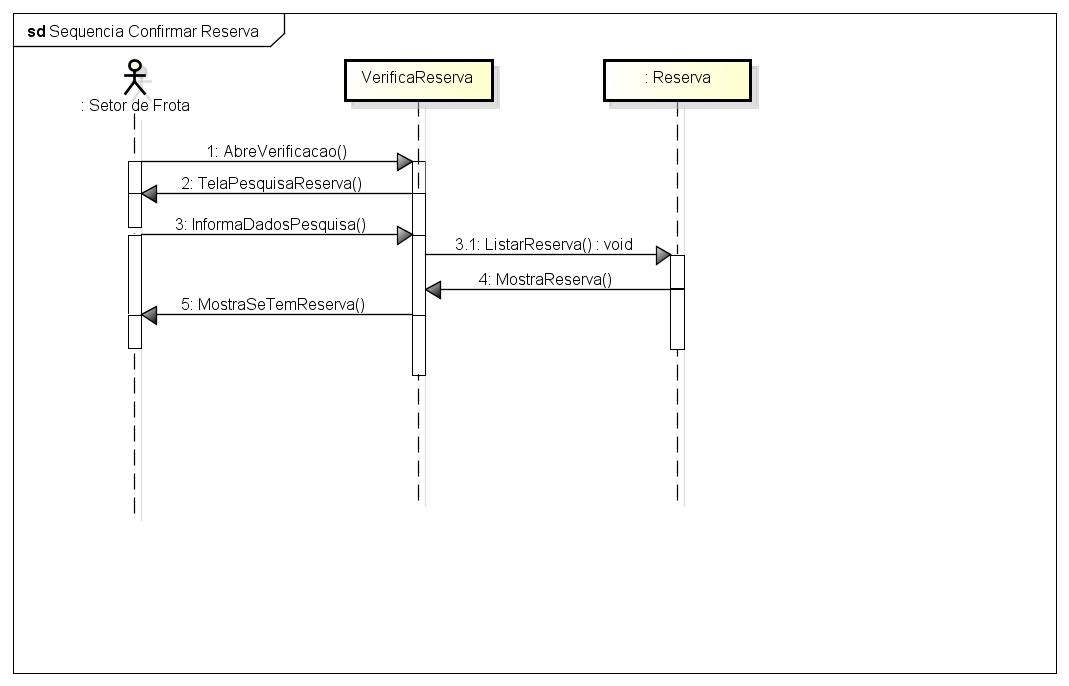
O Diagrama de Máquina de Estados representa as mudanças sofridas nos estados de uma instancia de uma classe ou caso de Uso.

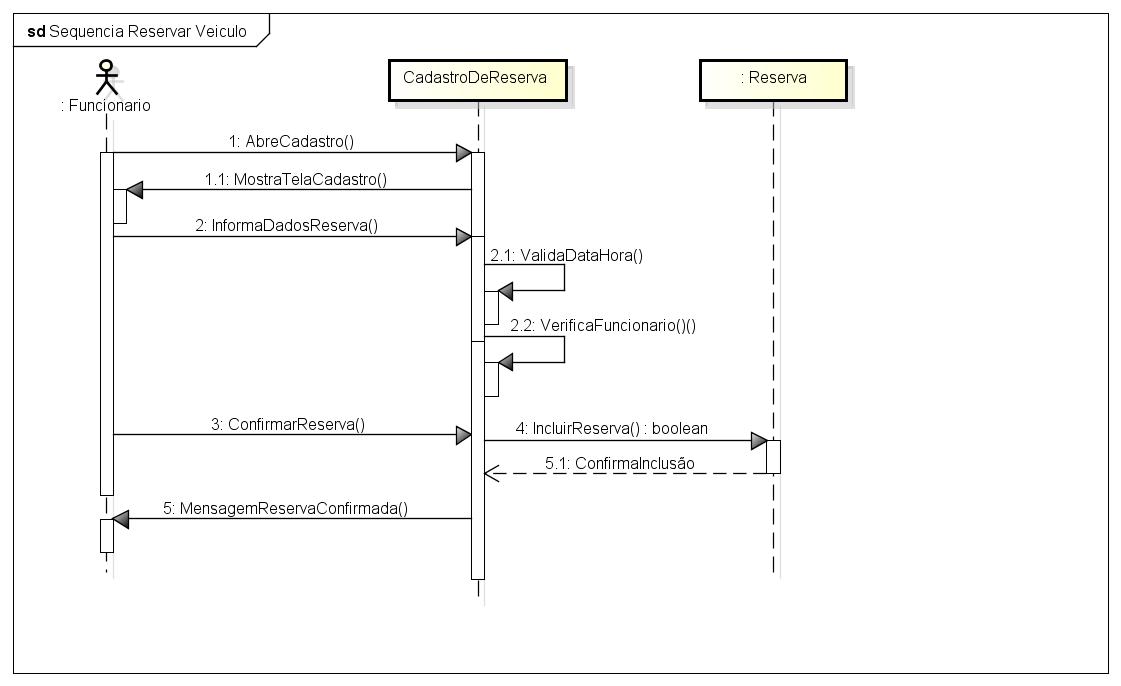


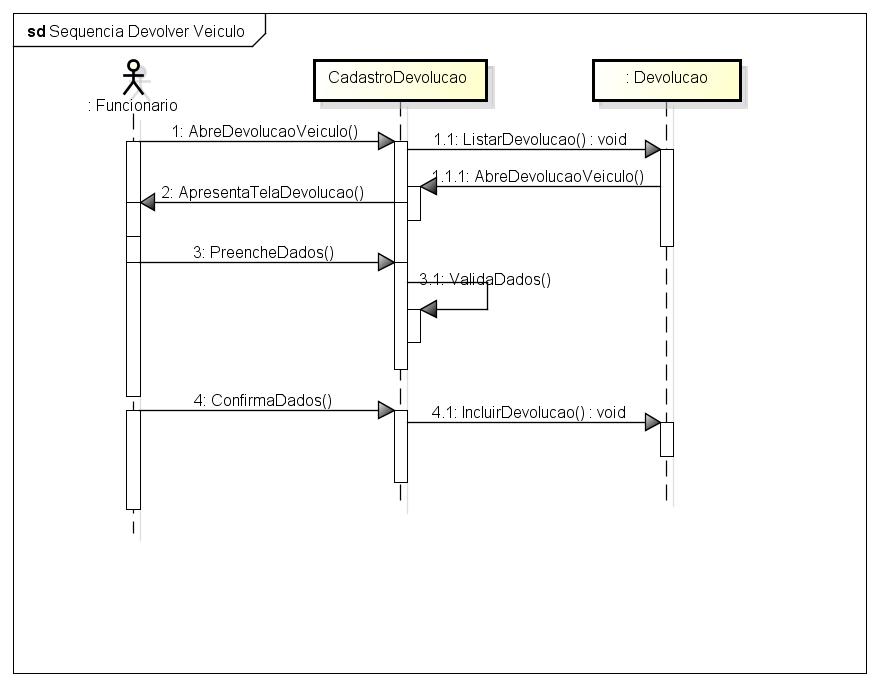
#### 3.2.4. Diagrama de Sequência:

O Diagrama de Sequência representa a ordem de tempo em que as mensagens são trocadas envolvendo um determinado tempo para cada processo.



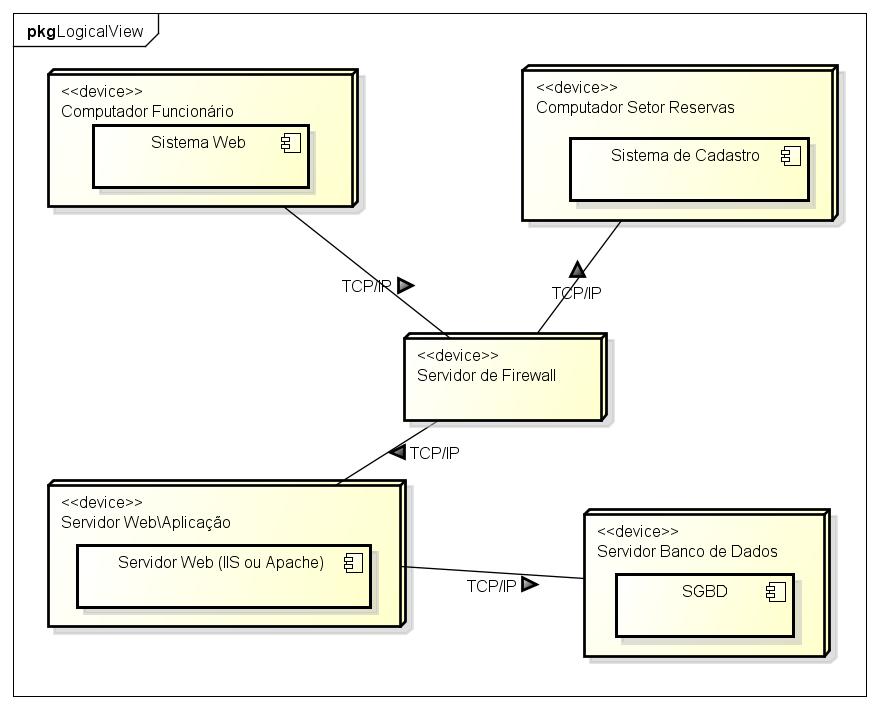






#### 3.2.5. Diagrama de Implantação:

O Diagrama de Implantação consiste em uma representação física, neste diagrama temos a representação dos relacionamentos entre software e hardware.



#### 3.3. BANCO DE DADOS.

No desenvolvimento de um novo sistema temos os requisitos a serem identificados onde é observado o projeto com suas características e relacionamentos, com todas as informações obtidas vamos desenvolver um modelo conceitual para definir o desenvolvimento do projeto.

#### 3.3.1. Modelo Relacional Normalizado:

O Modelo Relacional Normalizado é o processo usado para simplificar os dados de um banco de uma forma mais eficiente através da eliminação de dados redundantes eliminando.

Assim foram criadas algumas formas e normas para podermos eliminarmos os possíveis erros no projeto onde temos;

Primeira Forma Normal (1NF);

Segunda Forma Normal (2NF);

Terceira Forma Normal (3NF);

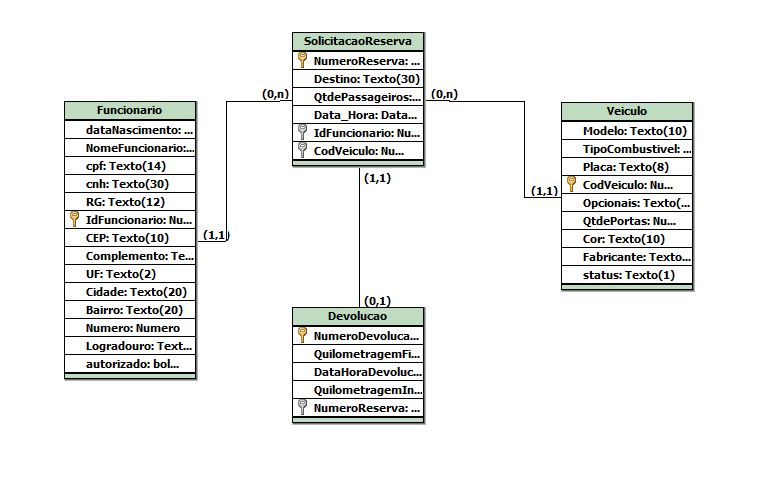
Boyce Codd;

Quarta Forma Normal (4NF) e;

Quinta Forma Normal (5NF).

Todas as formas normais são relacionadas, ou seja, um banco de dados está na 3NF, por definição ele também está nas formas normais 1NF e 2NF.

Na tela abaixo temos um Diagrama Entidade Relacionamento Normalizado, baseando em nosso estudo de caso.



#### 3.3.2. Modelo Entidade Relacionamento:

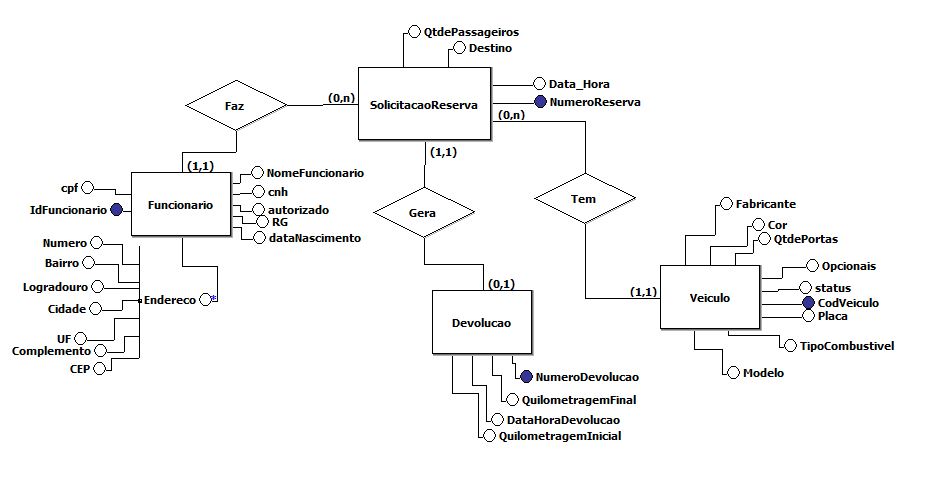
O MER (Modelo Entidade Relacionamento) é um modelo conceitual utilizado na Engenharia de Software que descreve os objetos envolvidos em um projeto de negócios, com seus atributos e como elas se relacionam entre si.

#### 3.3.3. Diagrama Entidade Relacionamento:

O DER (Diagrama Entidade Relacionamento) é uma ferramenta de desenvolvimento onde podemos visualizar o projeto graficamente e também é neste ponto que temos a visão exata de como se comportará o sistema.

Neste projeto gráfico temos que nos orientar por alguns padrões adotados pela proposta de Peter Chen.

As Entidades são representadas por Losangos; Os Atributos são representados por elipses; As Linhas ligam as entidades; As Cardinalidades ficam nas linhas.

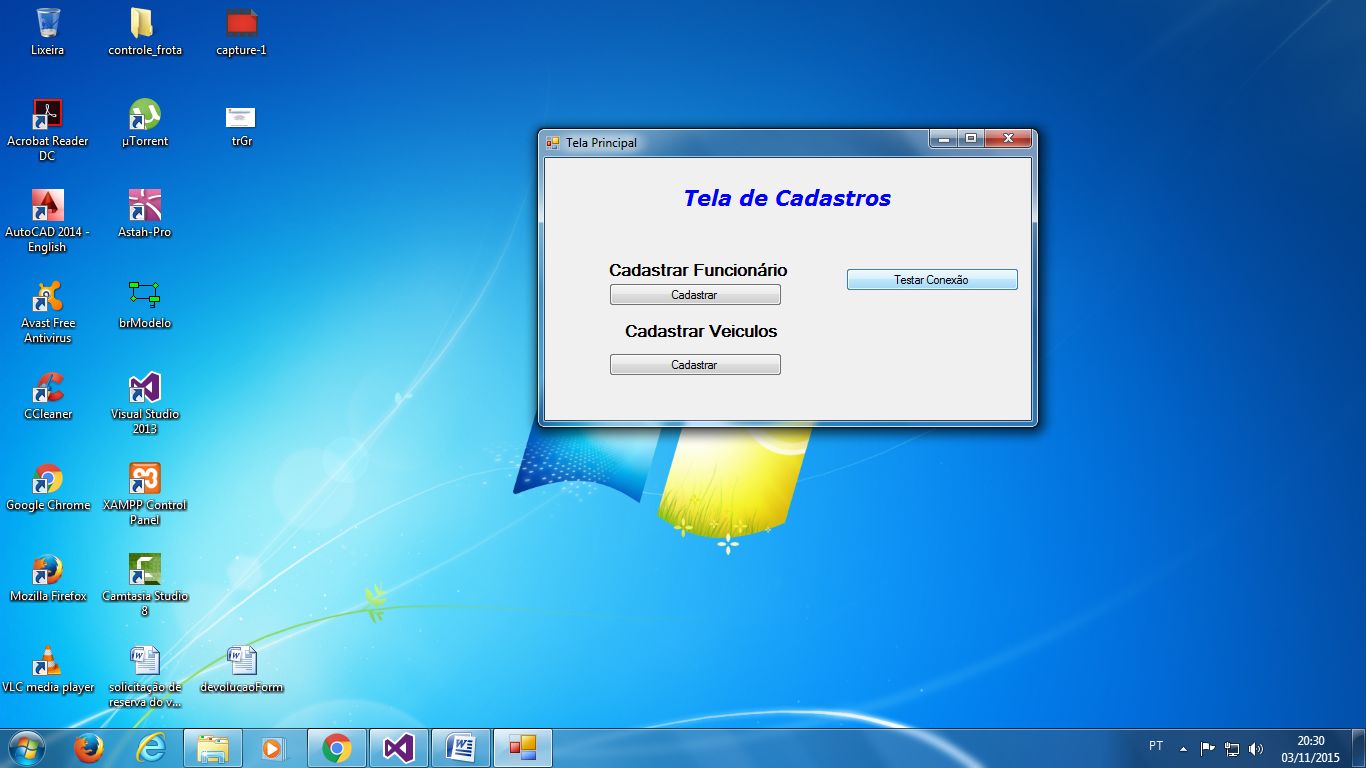
 Logo abaixo temos um exemplo do Diagrama de Entidade Relacionamento Normalizado Logico, também baseando em nosso estudo de Caso:

#### 3.4. PROGRAMAÇÃO ORIENTADA A OBJETOS.

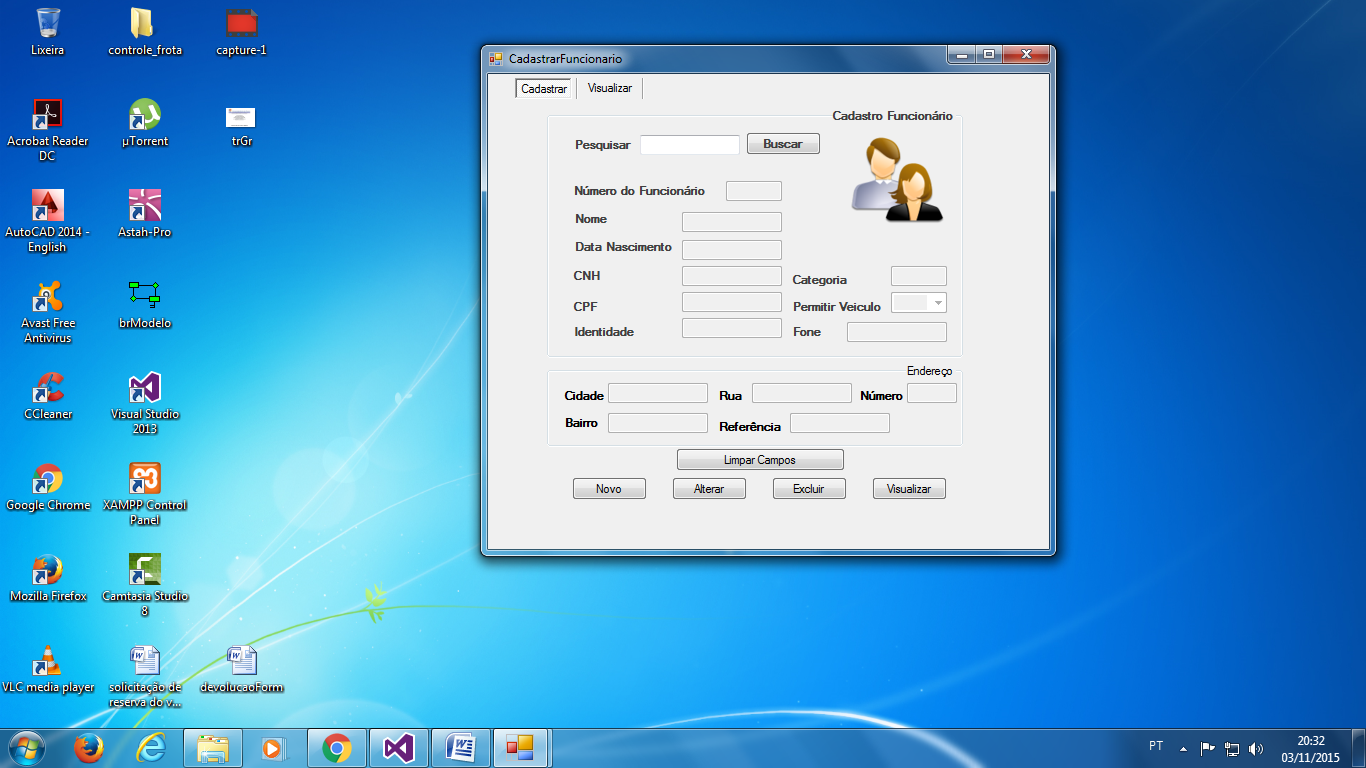
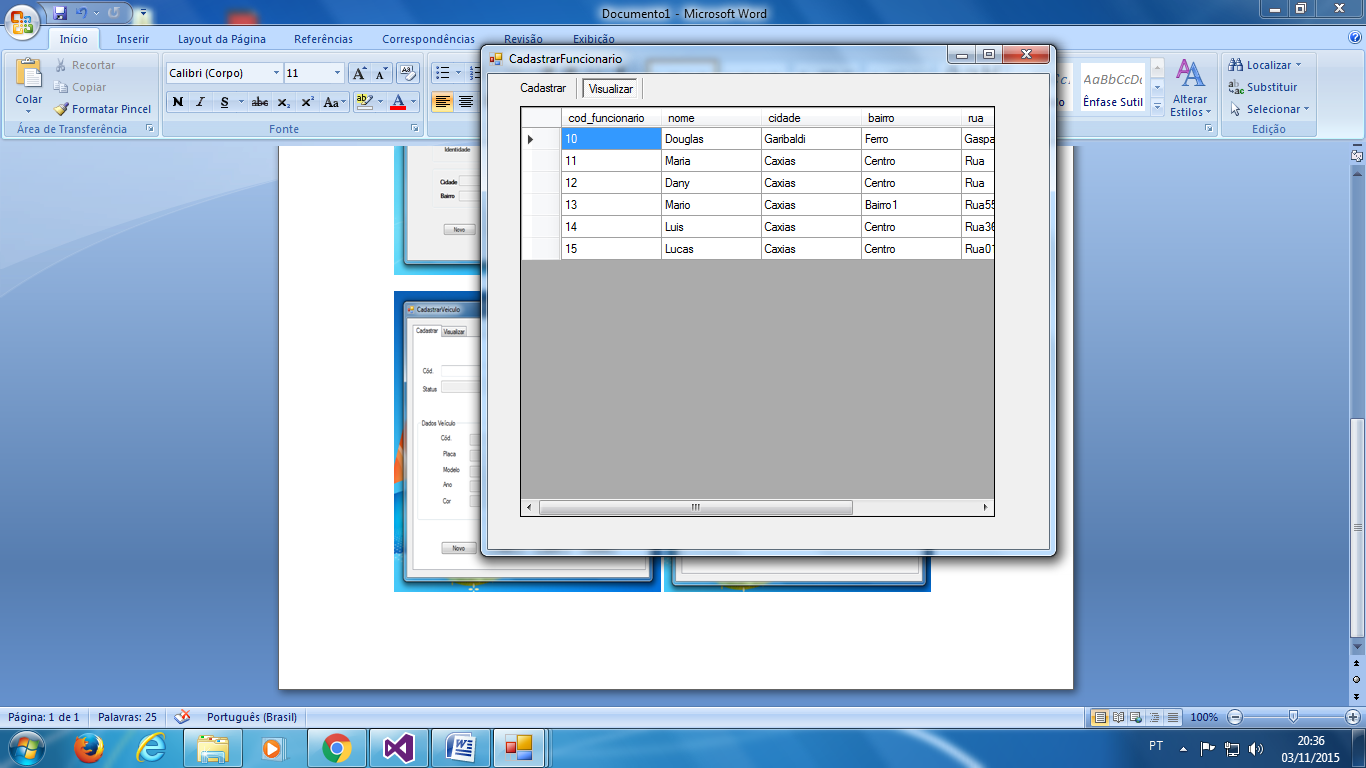
Na linguagem Orientada a Objetos, existe uma grande quantidade de IDE no mercado, como exemplo Java, C# e C++.

Neste trabalho usaremos o C# (C Sharp) para desenvolver o cadastro de funcionários e veículos, conforme as telas abaixo.

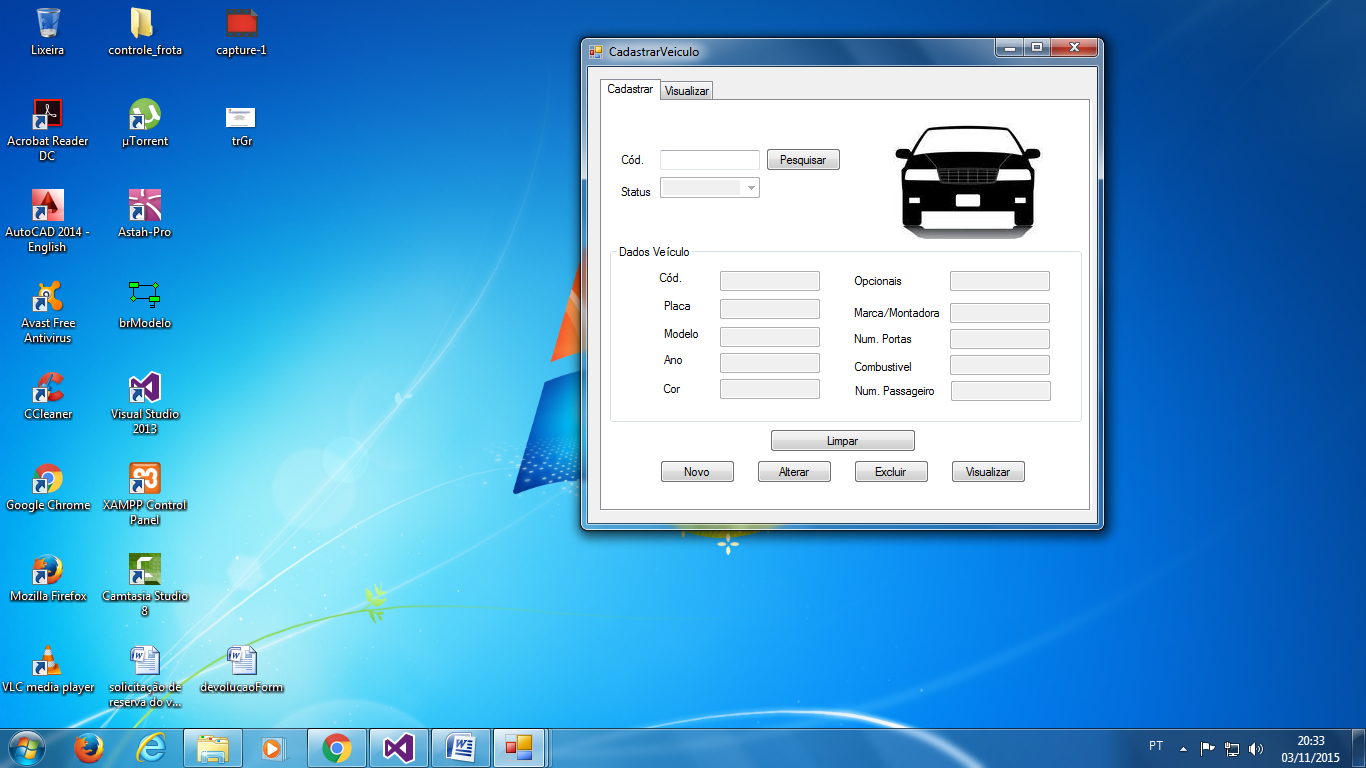
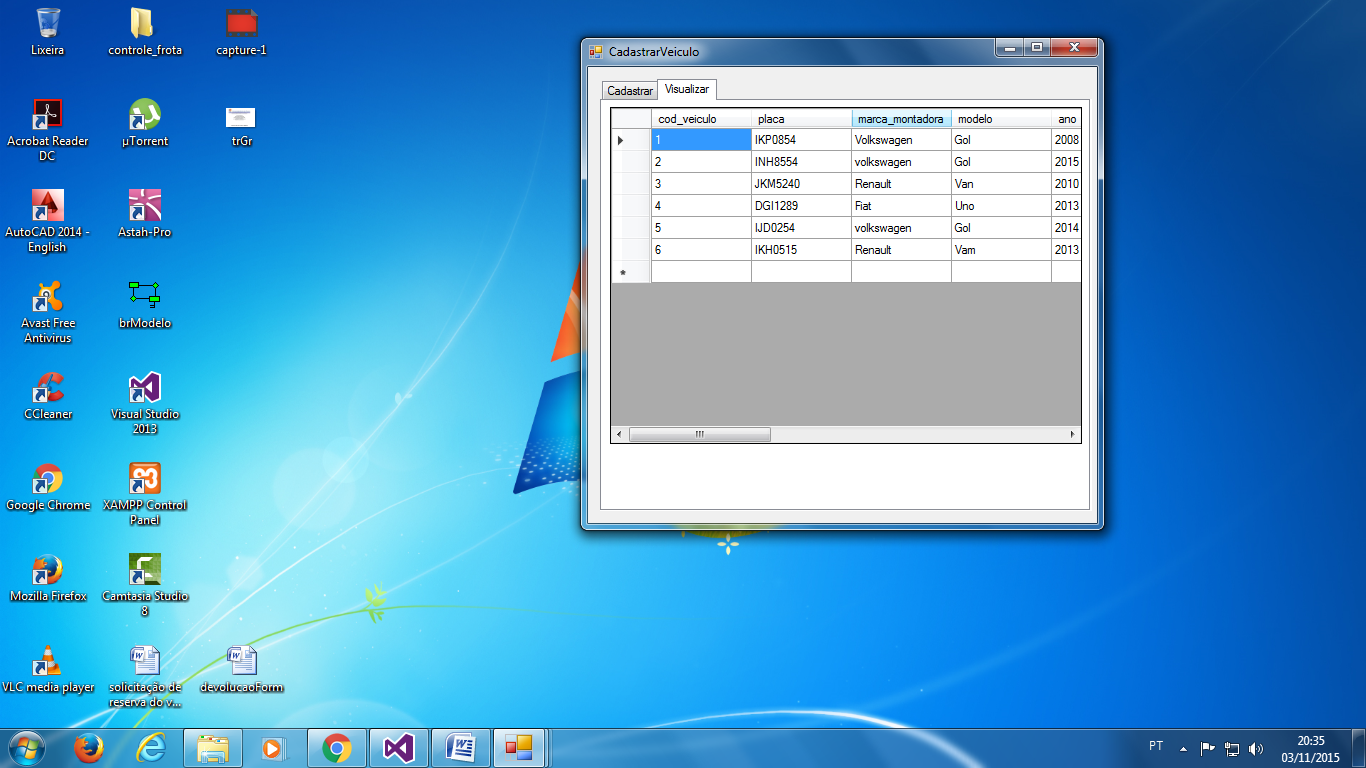
Segue abaixo uma sequência de 4 telas onde se dará o cadastro.



Nesta tela temos o Início dos cadastros



Nessas telas temos o cadastro de Funcionários



Nessas telas temos o cadastro de Veículos.

#### 3.4.1. Geração de Modelo físico.

Fragmentos de Codigos ultilizado em alguma das Telas.

Class Funcionario:

“ namespace PrincipalForms

{

class Funcionarios

{

private int codfuncionario;

private string nome;

private string datanasc;

private string fone;

private string permicao;

private string categoria\_cnh;

//Endeço

private string cidade;

private string bairro;

private string rua;

private string numero;

private string referencia;

//Documentos

private string numero\_cnh;

private string numero\_cpf;

private string numero\_ident;

public int CodFuncionario

{

get { return codfuncionario; }

set { codfuncionario = value; }

}

public string Nome

{

get { return nome; }

set { nome = value; }

}

public string DataNasc

{

get { return datanasc; }

set { datanasc = value; }

}

public string Fone

{

get { return fone; }

set { fone = value; }

}

public string Permicao

{

get { return permicao; }

set { permicao = value; }

}

public string Categoria\_cnh

{

get { return categoria\_cnh; }

set { categoria\_cnh = value; }

}

“

Classe de acesso ao Banco, FuncionarioDAL.

namespace PrincipalForms

{

class FuncionarioDAL

{

String conexaoMySQL = "";

public MySqlConnection con;

public FuncionarioDAL()

{

//Fazendo a Conexão com o Banco de Dados e Salvando na Variavel conexaoMySQL

conexaoMySQL = "Server=localHost;userid=root;password=;database=banco\_controlefrota";

}

//Seleciona a Tabela "Clientes" do Banco de Dados MySQL

public List<Funcionarios> selectLisFuncionarios(){

try {

using(MySqlConnection conn = new MySqlConnection(conexaoMySQL))

{

using (MySqlCommand command = new MySqlCommand("Select \* from funcionario, endereco\_funcionario", conn))

{

conn.Open();

List<Funcionarios> listaFuncionarios = new List<Funcionarios>();

using (MySqlDataReader dr = command.ExecuteReader())

{

while (dr.Read()) {

Funcionarios Funcionario = new Funcionarios();

Funcionario.CodFuncionario = (int)dr["cod\_funcionario"];

Funcionario.Nome = (string)dr["nome"];

Funcionario.DataNasc = (string)dr["data\_nasc"];

Funcionario.Fone = (string)dr["fone"];

Funcionario.Permicao = (string)dr["permitir"];

Funcionario.Categoria\_cnh = (string)dr["categoria\_cnh"];

//Endereço

Funcionario.Cidade = (string)dr["cidade"];

Funcionario.Bairro = (string)dr["bairro"];

Funcionario.Rua = (string)dr["rua"];

Funcionario.Numero = (string)dr["numero"];

Funcionario.Referencia = (string)dr["referencia"];

}

}

return listaFuncionarios;

}

}

}

catch (Exception ex) {

throw new Exception("Erro ao acessar Tabela Funcionario" + ex.Message);

}

}

// Select Tabela Clientes

public DataTable selectFuncionario()

{

try

{

String sql = "SELECT funcionario.cod\_funcionario, funcionario.nome, endereco\_funcionario.cidade, endereco\_funcionario.bairro, endereco\_funcionario.rua, endereco\_funcionario.numero FROM endereco\_funcionario, funcionario where funcionario.cod\_funcionario = endereco\_funcionario.Cod\_do\_funcionario";

con = new MySqlConnection(conexaoMySQL);

MySqlCommand cmd = new MySqlCommand(sql, con);

MySqlDataAdapter da = new MySqlDataAdapter();

da.SelectCommand = cmd;

DataTable dt = new DataTable();

da.Fill(dt);

return dt;

}

catch (Exception ex)

{

throw ex;

}

}

// Select Por Cod

public Funcionarios selectFuncionarioUltimoCod(int cod\_funcionario)

{

try

{

String sql = "Select MAX(cod\_funcionario) From funcionario";

con = new MySqlConnection(conexaoMySQL);

MySqlCommand cmd = new MySqlCommand(sql, con);

cmd.Parameters.AddWithValue("@cod\_funcionario", cod\_funcionario);

con.Open();

MySqlDataReader dr;

Funcionarios Funcionario = new Funcionarios();

dr = cmd.ExecuteReader(CommandBehavior.CloseConnection);

while (dr.Read())

{

Funcionario.CodFuncionario = Convert.ToInt32(dr["cod\_funcionario"]);

}

return Funcionario;

}

catch (Exception ex)

{

throw ex;

}

}

//Fazendo SELEC em Tabela

public Funcionarios selectFuncionarioByCod(int cod\_funcionario)

{

try

{

String sql = "SELECT \* FROM funcionario, endereco\_funcionario, doc\_funcionario WHERE funcionario.cod\_funcionario=@cod\_funcionario AND endereco\_funcionario.cod\_do\_funcionario=@cod\_funcionario AND doc\_funcionario.cod\_do\_funcionario=@cod\_funcionario" ;

con = new MySqlConnection (conexaoMySQL);

MySqlCommand cmd = new MySqlCommand(sql, con);

cmd.Parameters.AddWithValue("@cod\_funcionario", cod\_funcionario);

con.Open();

MySqlDataReader dr;

Funcionarios Funcionario = new Funcionarios();

dr = cmd.ExecuteReader(CommandBehavior.CloseConnection);

while (dr.Read())

{

Funcionario.CodFuncionario = Convert.ToInt32(dr["cod\_funcionario"]);

Funcionario.Nome = dr["nome"].ToString();

Funcionario.DataNasc = dr["data\_nasc"].ToString();

Funcionario.Fone = dr["fone"].ToString();

Funcionario.Permicao = dr["permitir"].ToString();

Funcionario.Categoria\_cnh = dr["categoria\_cnh"].ToString();

//Endereço

Funcionario.Cidade = dr["cidade"].ToString();

Funcionario.Rua = dr["rua"].ToString();

Funcionario.Bairro = dr["bairro"].ToString();

Funcionario.Referencia = dr["referecia"].ToString();

Funcionario.Numero = dr["numero"].ToString();

//DOC

Funcionario.Numero\_CNH = dr["CNH"].ToString();

Funcionario.Numero\_CPF = dr["CPF"].ToString();

Funcionario.Numero\_Ident = dr["num\_id"].ToString();

}

return Funcionario;

}

catch (Exception ex)

{

throw ex;

}

}

//Inserindo Dados

public void insertFuncionarios(Funcionarios Func)

{

try {

con = new MySqlConnection(conexaoMySQL);

con.Open();

String sql = ("Select MAX(cod\_funcionario) From funcionario");

MySqlCommand cm = new MySqlCommand(sql, con);

int codigo = 1 + (int)cm.ExecuteScalar();

string inserir\_func = "INSERT INTO funcionario(nome,data\_nasc,fone,permitir,categoria\_cnh)values('" + Func.Nome + "','" + Func.DataNasc + "','" + Func.Fone + "','" + Func.Permicao + "','" + Func.Categoria\_cnh + "')";

MySqlCommand comandos1 = new MySqlCommand(inserir\_func, con);

string inserir\_doc\_func = "INSERT INTO doc\_funcionario(cod\_do\_funcionario,CNH,CPF,num\_id)values('" + codigo + "','" + Func.Numero\_CNH + "','" + Func.Numero\_CPF + "','" + Func.Numero\_Ident + "')";

MySqlCommand comandos2 = new MySqlCommand(inserir\_doc\_func, con);

string inserir\_func\_End = "INSERT INTO endereco\_funcionario(cod\_do\_funcionario,cidade,rua,bairro,referecia,numero)values('" + codigo + "','" + Func.Cidade + "','" + Func.Rua + "','" + Func.Bairro + "','" + Func.Referencia + "','" + Func.Numero + "')";

MySqlCommand comandos3 = new MySqlCommand(inserir\_func\_End, con);

comandos1.ExecuteNonQuery();

comandos2.ExecuteNonQuery();

comandos3.ExecuteNonQuery();

con.Close();

}

catch (Exception ex)

{

throw new Exception("Erro de Comandos ->Inserindo dados<- :" + ex.Message);

}

}

//Atualizando

public void updateFuncionario(Funcionarios Funcionario)

{

try

{

String sql1 = "UPDATE endereco\_funcionario SET cidade=@cidade, rua=@rua,bairro=@bairro,referecia=@referencia,numero=@numero WHERE cod\_do\_funcionario=@cod\_funcionario";

String sql2 = "UPDATE doc\_funcionario SET CNH=@CNH, CPF=@CPF,num\_id=@num\_id WHERE cod\_do\_funcionario=@cod\_funcionario";

String sql3 = "UPDATE funcionario SET nome=@nome, data\_nasc=@data\_nasc, fone=@fone,permitir=@permitir,categoria\_cnh=@categoria\_cnh WHERE cod\_funcionario=@cod\_funcionario";

con = new MySqlConnection(conexaoMySQL);

MySqlCommand cmd1 = new MySqlCommand(sql1, con);

MySqlCommand cmd2 = new MySqlCommand(sql2, con);

MySqlCommand cmd3 = new MySqlCommand(sql3, con);

cmd1.Parameters.AddWithValue("@cod\_funcionario", Funcionario.CodFuncionario);

cmd1.Parameters.AddWithValue("@cidade", Funcionario.Cidade);

cmd1.Parameters.AddWithValue("@rua", Funcionario.Rua);

cmd1.Parameters.AddWithValue("@bairro", Funcionario.Bairro);

cmd1.Parameters.AddWithValue("@referencia", Funcionario.Referencia);

cmd1.Parameters.AddWithValue("@numero", Funcionario.Numero);

cmd2.Parameters.AddWithValue("@cod\_funcionario", Funcionario.CodFuncionario);

cmd2.Parameters.AddWithValue("@CNH", Funcionario.Numero\_CNH);

cmd2.Parameters.AddWithValue("@CPF", Funcionario.Numero\_CPF);

cmd2.Parameters.AddWithValue("@num\_id", Funcionario.Numero\_Ident);

cmd3.Parameters.AddWithValue("@cod\_funcionario", Funcionario.CodFuncionario);

cmd3.Parameters.AddWithValue("@nome", Funcionario.Nome);

cmd3.Parameters.AddWithValue("@data\_nasc", Funcionario.DataNasc);

cmd3.Parameters.AddWithValue("@fone", Funcionario.Fone);

cmd3.Parameters.AddWithValue("@permitir", Funcionario.Permicao);

cmd3.Parameters.AddWithValue("@categoria\_cnh", Funcionario.Categoria\_cnh);

con.Open();//abri conexão com banco

cmd1.ExecuteNonQuery();

cmd2.ExecuteNonQuery();

cmd3.ExecuteNonQuery();

}

catch (Exception ex)

{

throw ex;

}

finally

{

con.Close();

}

}

//Deletando

public void deleteFuncionario(Funcionarios funcionario)

{

try

{

MySqlConnection con = new MySqlConnection(conexaoMySQL);

String sql1 = "DELETE FROM endereco\_funcionario WHERE cod\_do\_funcionario = @cod\_funcionario ";

String sql2 = "DELETE FROM doc\_funcionario WHERE cod\_do\_funcionario = @cod\_funcionario ";

String sql3 = "DELETE FROM funcionario WHERE cod\_funcionario = @cod\_funcionario";

MySqlCommand cmd1 = new MySqlCommand(sql1, con);

cmd1.Parameters.AddWithValue("@cod\_funcionario", funcionario.CodFuncionario);

MySqlCommand cmd2 = new MySqlCommand(sql2, con);

cmd2.Parameters.AddWithValue("@cod\_funcionario", funcionario.CodFuncionario);

MySqlCommand cmd3 = new MySqlCommand(sql3, con);

cmd3.Parameters.AddWithValue("@cod\_funcionario", funcionario.CodFuncionario);

con.Open();

cmd1.ExecuteNonQuery();

cmd2.ExecuteNonQuery();

cmd3.ExecuteNonQuery();

con.Close();

}catch (Exception ex)

{

throw new Exception("Erro de Comandos-> Delet: " + ex.Message);

}

}

public MySqlConnection conex{ get; set; }

}

}

Classe FuncionarioBLL.

class FuncionarioBLL

{

FuncionarioDAL dal = null;

public FuncionarioBLL()

{

}

//

public DataTable selecionaFuncionario()

{

DataTable tb = new DataTable();

try {

dal = new FuncionarioDAL();

tb = dal.selectFuncionario();

}

catch(Exception ex)

{

throw ex;

}

return tb;

}

//

public void novoFuncionario(Funcionarios Func)

{

try

{

dal = new FuncionarioDAL();

dal.insertFuncionarios(Func);

}

catch (Exception ex)

{

throw ex;

}

}

//

public void alterarFuncionario(Funcionarios Funcionario)

{

try

{

dal = new FuncionarioDAL();

dal.updateFuncionario(Funcionario);

}

catch (Exception ex)

{

throw ex;

}

}

//

public void excluirFuncionario(Funcionarios funcionario)

{

try

{

dal = new FuncionarioDAL();

dal.deleteFuncionario(funcionario);

}

catch (Exception ex)

{

throw ex;

}

}

//

public List<Funcionarios> listaFuncionarios()

{

try

{

dal = new FuncionarioDAL();

return dal.selectLisFuncionarios();

}

catch (Exception ex)

{

throw ex;

}

}

//

public Funcionarios listaFuncionarioPorCod(int cod\_funcionario)

{

try

{

dal = new FuncionarioDAL();

return dal.selectFuncionarioByCod(cod\_funcionario);

}

catch (Exception ex)

{

throw ex;

}

}

}

Class CadastrarFuncionario : Form.

public partial class CadastrarFuncionario : Form

{

public CadastrarFuncionario()

{

InitializeComponent();

}

private void btnVisualizar\_Click(object sender, EventArgs e)

{

try

{

FuncionarioBLL bll = new FuncionarioBLL();

dgvFuncionario.DataSource = bll.selecionaFuncionario();

}

catch (Exception ex)

{

MessageBox.Show("Erro:" + ex.Message.ToString());

}

}

private void btnBuscar\_Click(object sender, EventArgs e)

{

if ((txtPesquisa.TextLength) > 0)

{

int cod\_funcionario = Convert.ToInt32(txtPesquisa.Text);

Funcionarios c = new Funcionarios();

if ((txtPesquisa.TextLength) > 0)

btnSalvar.Enabled = false;

btnExcluir.Enabled = true;

btnAlterar.Enabled = true;

txtNome.Enabled = true;

txtDataNasc.Enabled = true;

txtFone.Enabled = true;

txtPermicao.Enabled = true;

txtCategoria.Enabled = true;

txtCidade.Enabled = true;

txtBairro.Enabled = true;

txtRua.Enabled = true;

txtNum.Enabled = true;

txtReferencia.Enabled = true;

txtIdentidade.Enabled = true;

txtCNH.Enabled = true;

txtCPF.Enabled = true;

try

{

FuncionarioBLL bll = new FuncionarioBLL();

c = bll.listaFuncionarioPorCod(cod\_funcionario);

txtCodFuncionario.Text = c.CodFuncionario.ToString();

txtNome.Text = c.Nome;

txtDataNasc.Text = c.DataNasc;

txtFone.Text = c.Fone;

txtPermicao.Text = c.Permicao;

txtCategoria.Text = c.Categoria\_cnh;

txtCidade.Text = c.Cidade;

txtBairro.Text = c.Bairro;

txtRua.Text = c.Rua;

txtNum.Text = c.Numero;

txtReferencia.Text = c.Referencia;

txtIdentidade.Text = c.Numero\_Ident;

txtCNH.Text = c.Numero\_CNH;

txtCPF.Text = c.Numero\_CPF;

}

catch (Exception ex)

{

MessageBox.Show(" Erro : " + ex.Message.ToString());

}

}

else

{

MessageBox.Show("Digite um Código de Funcionario!");

txtPesquisa.Text = "1";

}

}

private void btnExcluir\_Click(object sender, EventArgs e)

{

Funcionarios c = new Funcionarios();

if ((txtCodFuncionario.TextLength) <= 0)

{

MessageBox.Show("Sem Dados para ser Excluir! Faça uma Pesquisa...");

txtCodFuncionario.Enabled = false;

txtNome.Enabled = false;

txtDataNasc.Enabled = false;

txtFone.Enabled = false;

txtPermicao.Enabled = false;

txtCategoria.Enabled = false;

txtCidade.Enabled = false;

txtBairro.Enabled = false;

txtRua.Enabled = false;

txtNum.Enabled = false;

txtReferencia.Enabled = false;

txtIdentidade.Enabled = false;

txtCNH.Enabled = false;

txtCPF.Enabled = false;

}

else

{

c.CodFuncionario = Convert.ToInt32(txtCodFuncionario.Text);

}

try

{

FuncionarioBLL bll = new FuncionarioBLL();

bll.excluirFuncionario(c);

limparTextBoxs(this);

MessageBox.Show(" Excluido!");

}

catch (Exception ex)

{

MessageBox.Show(" Erro : Excl" + ex.Message.ToString());

}

}

private void btnSalvar\_Click(object sender, EventArgs e)

{

txtCodFuncionario.Text = "";

if (txtNome.Enabled == false)//Deixa Campos enbled True

{

txtPesquisa.Text = ".";

txtCodFuncionario.Text = ".";

txtCodFuncionario.Enabled = false;

txtPesquisa.Enabled = false;

btnBuscar.Enabled = false;

btnAlterar.Enabled = false;

btnExcluir.Enabled = false;

txtNome.Enabled = true;

txtDataNasc.Enabled = true;

txtFone.Enabled = true;

txtPermicao.Enabled = true;

txtCategoria.Enabled = true;

txtCidade.Enabled = true;

txtBairro.Enabled = true;

txtRua.Enabled = true;

txtNum.Enabled = true;

txtReferencia.Enabled = true;

txtIdentidade.Enabled = true;

txtCNH.Enabled = true;

txtCPF.Enabled = true;

}

else

{

verificaCamposTextBoxs(this);

if (((txtNome.TextLength) > 0) && ((txtDataNasc.TextLength) > 0) && ((txtFone.TextLength) > 0) && ((txtReferencia.TextLength) > 0))

{

Funcionarios c = new Funcionarios();

c.Nome = (String)txtNome.Text;

c.DataNasc = (String)txtDataNasc.Text;

c.Fone = (String)txtFone.Text;

c.Permicao = (String)txtPermicao.Text;

c.Categoria\_cnh = (String)txtCategoria.Text;

c.Cidade = (String)txtCidade.Text;

c.Bairro = (String)txtBairro.Text;

c.Rua = (String)txtRua.Text;

c.Numero = (String)txtNum.Text;

c.Referencia = (String)txtReferencia.Text;

c.Numero\_CNH = (String)txtCNH.Text;

c.Numero\_CPF = (String)txtCPF.Text;

c.Numero\_Ident = (String)txtIdentidade.Text;

limparTextBoxs(this);

try

{

if(((txtCodFuncionario.TextLength) > 0))

MessageBox.Show("Não pode ter Código no Campo Número Funcionario"); limparTextBoxs(this);

FuncionarioBLL bll = new FuncionarioBLL();

bll.novoFuncionario(c);

MessageBox.Show("Salvos com Sucesso!");

}

catch (Exception ex)

{

MessageBox.Show(" Erro em Salvar: " + ex.Message.ToString());

}

}

}

}

private void limparTextBoxs(Control controle)

{

foreach (Control Ctle in controle.Controls)

{

if (Ctle is TextBox)

{

((TextBox)Ctle).Text = string.Empty;

}

else if (Ctle.Controls.Count > 0)

{

limparTextBoxs(Ctle);

}

}

}

private void verificaCamposTextBoxs(Control controle)

{

foreach (Control Ctle in controle.Controls)

{

if (Ctle is TextBox && Ctle.Text =="")

{

MessageBox.Show("Todos os Campos são Obrigatorios!");

return;

}

else if (Ctle.Controls.Count > 0)

{

verificaCamposTextBoxs(Ctle);

}

}

}

private void btnLimpar\_Click(object sender, EventArgs e)

{

limparTextBoxs(this);

btnSalvar.Enabled = true;

btnExcluir.Enabled = false;

btnAlterar.Enabled = false;

txtNome.Enabled = false;

txtDataNasc.Enabled = false;

txtFone.Enabled = false;

txtPermicao.Enabled = false;

txtCategoria.Enabled = false;

txtCidade.Enabled = false;

txtBairro.Enabled = false;

txtRua.Enabled = false;

txtNum.Enabled = false;

txtReferencia.Enabled = false;

txtIdentidade.Enabled = false;

txtCNH.Enabled = false;

txtCPF.Enabled = false;

}

private void btnAlterar\_Click(object sender, EventArgs e)

{

Funcionarios c = new Funcionarios();

if ((txtCodFuncionario.TextLength) > 0)

{

c.CodFuncionario = Convert.ToInt32(txtCodFuncionario.Text);

c.Nome = (string)txtNome.Text;

c.DataNasc = (string)txtDataNasc.Text;

c.Fone = (string)txtFone.Text;

c.Permicao = (string)txtPermicao.Text;

c.Categoria\_cnh = (string)txtCategoria.Text;

c.Numero\_CNH = (string)txtCNH.Text;

c.Numero\_CPF = (string)txtCPF.Text;

c.Numero\_Ident = (string)txtIdentidade.Text;

c.Cidade = (string)txtCidade.Text;

c.Rua = (string)txtRua.Text;

c.Bairro = (string)txtBairro.Text;

c.Numero = (string)txtNum.Text;

c.Referencia = (string)txtReferencia.Text;

limparTextBoxs(this);

try

{

FuncionarioBLL bll = new FuncionarioBLL();

bll.alterarFuncionario(c);

}

catch (Exception ex)

{

MessageBox.Show(" Erro : " + ex.Message.ToString());

}

}

else

{

MessageBox.Show("Para fazer Alteração, você deve Fazer uma Pesquisa!");

}

}

}

#### 3.5. PROGRAMAÇÃO WEB.

As Necessidades dos projetos desenvolvidos aplicam-se um a tecnologia de desenvolvimento Web que melhor atende o projeto.

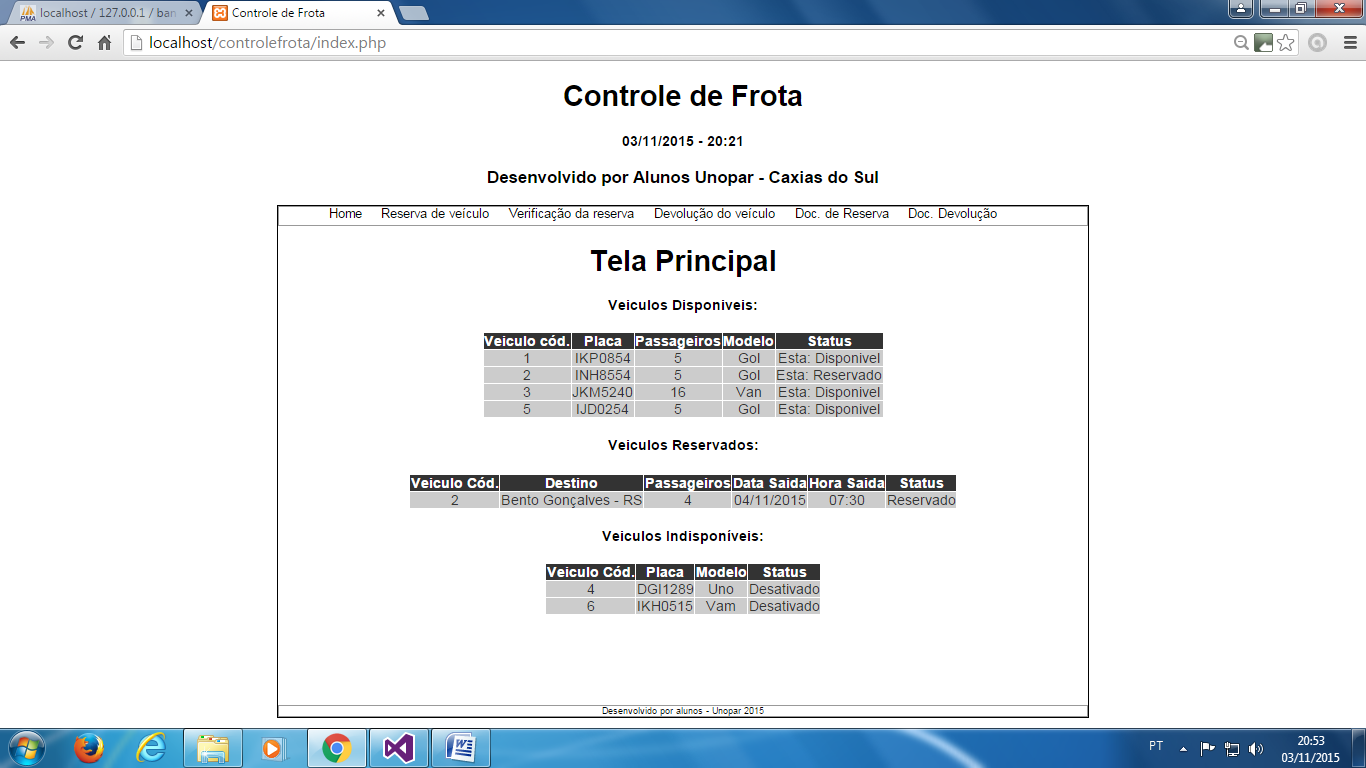
Para este trabalho vamos trabalhar com PHP:

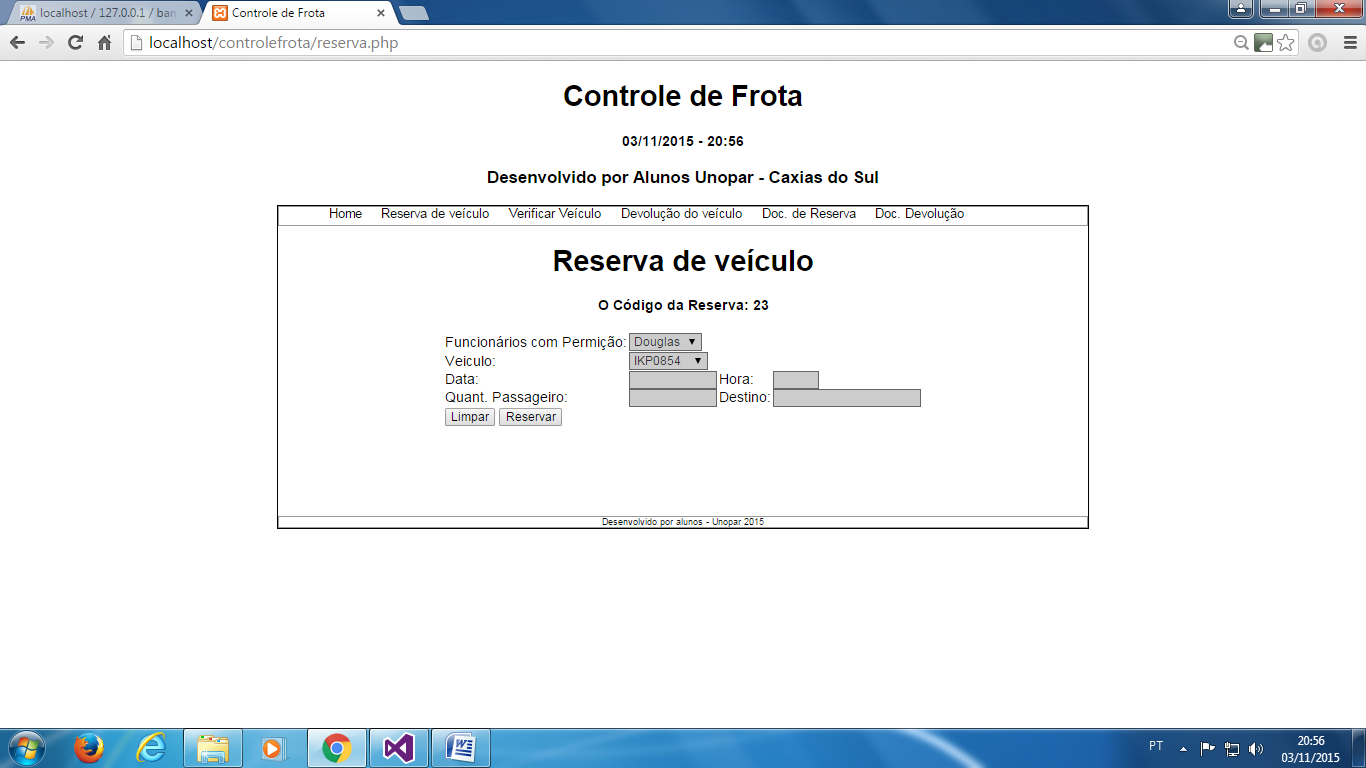
#### 3.5.1. Hypertext Preprocessor o PHP:

A sigla PHP significa Pré-Processador de Hipertexto. O PHP é uma linguagem de programação para criar documentos HTML, possui interação multiplataforma podendo rodar em Linux e Windows.

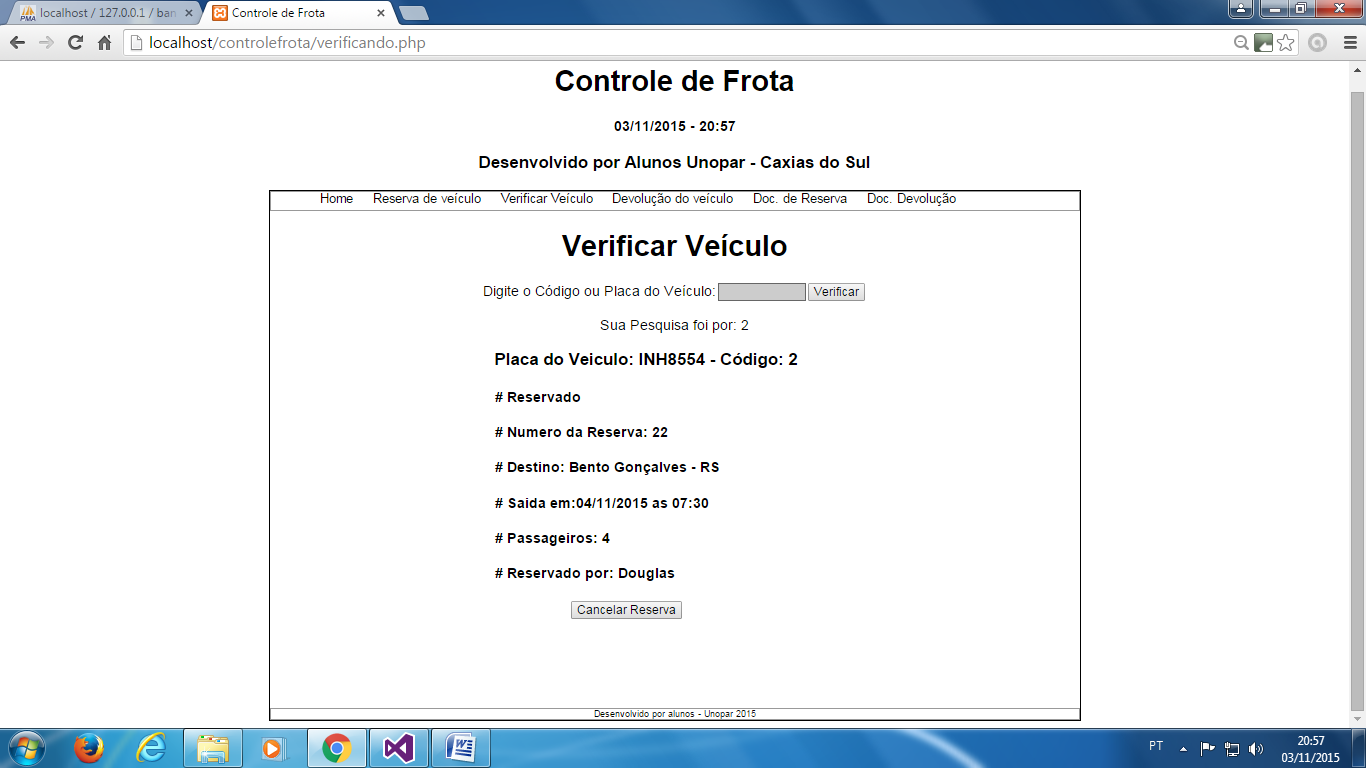
O PHP pode ser executado em vários servidores de Internet como o Apache e o ISS também é uma linguagem de última geração com características provenientes de Java, C++, PERL e C segundo pesquisas o PHP foi desenhado especialmente para a programação Web.

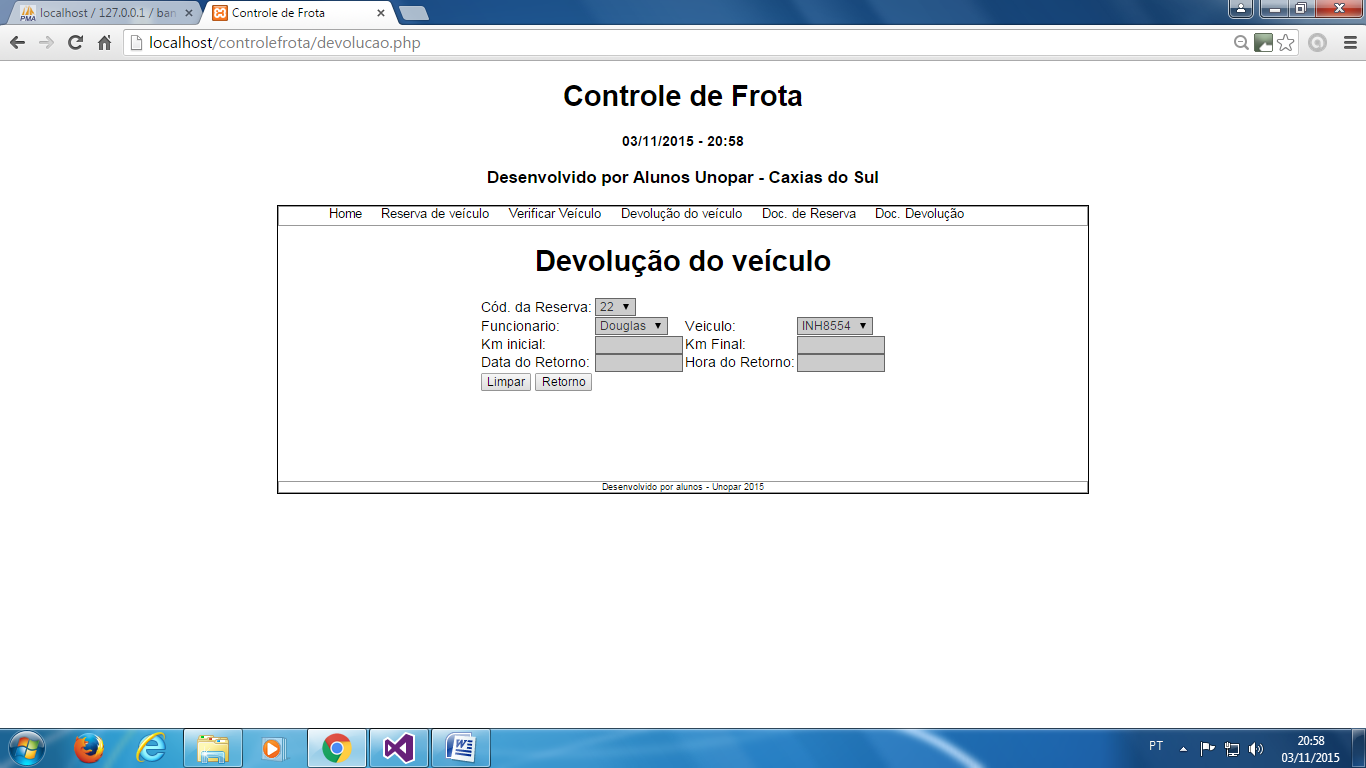
Segue abaixo a sequência de 4 telas onde teremos a reserva dos veículos.

Tela de Veículos Disponíveis.



Tela de Reserva de Veículos.

Tela com a Disponibilidade do Veiculo



Tela para devolução do Veiculo

#### 3.5.2. Geração de códigos Web.

Vamos mostrar a seguir fragmentros de códigos correspondente a desenvolvimento Web.

Pagina Index.php.

“<?php

require\_once "includes/config.php";

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Controle de Frota</title>

<link rel="stylesheet" type="text/css" href="css/stylo.css" />

</head>

<body>

<?php

require\_once "includes/cabeca.php";

?>

<div id="principal">

<div id="menu">

<nav>

<ul>

<li><a href="index.php" title="Inicio">Home</a></li>

<li><a href="reserva.php" title="Reserva de veículo">Reserva de veículo</a></li>

<li><a href="verifica.php" title="Verificação da reserva">Verificação da reserva</a></li>

<li><a href="devolucao.php" title="Devolução do veículo" >Devolução do veículo</a></li>

<li><a href="doc/soliReservVeic.pdf" target="\_blank">Doc. de Reserva</a></li>

<li><a href="doc/devolucaoForm.pdf" target="\_blank">Doc. Devolução</a></li>

</ul>

</nav>

</div>

<div id="centro">

<h1>Tela Principal</h1>

<H4>Veiculos Disponiveis:</H4>

<table id="tabela">

<tr class="form\_campos\_cab">

<td><strong>Veiculo cód.</strong></td>

<td><strong>Placa</strong></td>

<td><strong>Passageiros</strong></td>

<td><strong>Modelo</strong></td>

<td><strong>Status</strong></td>

</tr>

<?php

$SQL\_s = mysql\_query("SELECT \* FROM status\_veiculo WHERE status='Disponivel' or status='Reservado'");

while($res = mysql\_fetch\_array($SQL\_s)){

$cod\_veiculo = $res['cod\_veiculo'];

$statusV = $res['status'];

$SQL\_FN = mysql\_query("SELECT \* FROM veiculo WHERE cod\_veiculo='$cod\_veiculo' ORDER BY cod\_veiculo DESC");

while($ftn = mysql\_fetch\_array($SQL\_FN)){

?>

<article>

<tr class="form\_campos">

<td><?php echo $ftn['cod\_veiculo']; ?></td>

<td><?php echo $ftn['placa']; ?></td>

<td><?php echo $ftn['num\_passageiros']; ?></td>

<td><?php echo $ftn['modelo']; ?></td>

<td><?php echo "Esta: ".$statusV; ?></td>

</tr>

</article>

<?php } }?>

</table>

<table id="tabela" border="0">

<tr><H4>Veiculos Reservados:</H4></tr>

<tr class="form\_campos\_cab">

<td><strong>Veiculo Cód.</strong></td>

<td><strong>Destino</strong></td>

<td><strong>Passageiros</strong></td>

<td><strong>Data Saida</strong></td>

<td><strong>Hora Saida</strong></td>

<td><strong>Status</strong></td>

</tr>

<?php

date\_default\_timezone\_set('America/Sao\_Paulo');

$data = date('d/m/Y');

$hora = date('H:i');

$SQL\_FN = mysql\_query("SELECT \* FROM veiculo WHERE cod\_veiculo='$cod\_veiculo'");

while($ftn = mysql\_fetch\_array($SQL\_FN)){

$cod\_veiculo = $ftn['cod\_veiculo'];

$SQL\_R = mysql\_query("SELECT \* FROM reserva\_veiculo WHERE status='Reservado'");

while($res = mysql\_fetch\_array($SQL\_R)){

$hora = $res['hora\_saida'];

$destino = $res['destino'];

$passageiros = $res['num\_passageiros'];

$saida = $res['data\_saida'];

$statusR = $res['status'];

?>

<article>

<tr class="form\_campos">

<td><?php echo $res['cod\_veiculo'];?></td>

<td><?php echo $destino; ?></td>

<td><?php echo $passageiros; ?></td>

<td><?php echo $saida; ?></td>

<td><?php echo $hora; ?></td>

<td><?php echo $statusR; ?></td>

</tr>

</article>

<?php } }?>

</table>

<H4>Veiculos Indisponíveis:</H4>

<table id="tabela">

<tr class="form\_campos\_cab">

<td><strong>Veiculo Cód.</strong></td>

<td><strong>Placa</strong></td>

<td><strong>Modelo</strong></td>

<td><strong>Status</strong></td>

</tr>

<?php

$SQL\_E = mysql\_query("SELECT \* FROM status\_veiculo WHERE status='Em Manutencao' OR status='Desativado' OR status='Indisponivel'");

while($ec = mysql\_fetch\_array($SQL\_E)){

$cod\_veiculo = $ec['cod\_veiculo'];

$status = $ec['status'];

$string= mb\_convert\_encoding(utf8\_encode($status), 'UTF-8');

$SQL\_FN = mysql\_query("SELECT \* FROM veiculo WHERE cod\_veiculo='$cod\_veiculo' ORDER BY cod\_veiculo DESC");

while($ftn = mysql\_fetch\_array($SQL\_FN)){

?>

<article>

<tr class="form\_campos">

<td><?php echo $ftn['cod\_veiculo']; ?></td>

<td><?php echo $ftn['placa']; ?></td>

<td><?php echo $ftn['modelo']; ?></td>

<td><?php echo $string; ?></td>

</tr>

</article>

<?php } }?>

</table>

</div>

<div id="rodape">

Desenvolvido por alunos - Unopar 2015

</div>

</div>

</body>

</html>

”

Fragmento da pagina reserva.php.

“<?php

require\_once "includes/config.php";

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Controle de Frota</title>

<link rel="stylesheet" type="text/css" href="css/stylo.css" />

</head>

<body>

<?php

require\_once "includes/cabeca.php"

?>

<div id="principal">

<div id="menu">

<nav>

<ul>

<li><a href="index.php" title="Inicio">Home</a></li>

<li><a href="reserva.php" title="Reserva de veículo">Reserva de veículo</a></li>

<li><a href="verifica.php" title="Verificar Veículo">Verificar Veículo</a></li>

<li><a href="devolucao.php" title="Devolução do veículo" >Devolução do veículo</a></li>

<li><a href="doc/soliReservVeic.pdf" target="\_blank">Doc. de Reserva</a></li>

<li><a href="doc/devolucaoForm.pdf" target="\_blank">Doc. Devolução</a></li>

</ul>

</nav>

</div>

<div id="centro">

<h1>Reserva de veículo</h1>

<h4>

<?php

$SQL\_D = mysql\_query("SELECT cod\_reserva FROM reserva\_veiculo ORDER BY cod\_reserva ");

while($re = mysql\_fetch\_array($SQL\_D)){

$cod\_reser = $re['cod\_reserva'];

$cod\_reserv = $cod\_reser + 1;

}

echo "O Código da Reserva: ".$cod\_reserv;

?></h4>

<form action="acoes/reservar.php" name="formulario" method="post" enctype="multipart/form-data">

<table id="tabela" border="0">

<tr>

<td>

Funcionários com Permição: </td><td>

<select class="form\_campos" name="codigoFuncionario">

<?php

$SQL\_F = mysql\_query("SELECT \* FROM funcionario WHERE permitir='Sim' ");

while($func = mysql\_fetch\_array($SQL\_F)){

$cod\_func = $func['cod\_funcionario'];

$nome = $func['nome'];

?>

<option value="<?php echo $cod\_func; ?>"><?php echo $nome; ?></option>

<?php }

?>

</select>

</td></tr><tr><td>

Veiculo: </td><td>

<select class="form\_campos" name="codigoVeiculo">

<?php

$SQL\_E = mysql\_query("SELECT \* FROM status\_veiculo WHERE status='Disponivel' or status='Reservado'");

while($ec = mysql\_fetch\_array($SQL\_E)){

$cod\_veiculo = $ec['cod\_veiculo'];

$status = $ec['status'];

$SQL\_FN = mysql\_query("SELECT \* FROM veiculo WHERE cod\_veiculo='$cod\_veiculo' ORDER BY cod\_veiculo DESC");

while($ftn = mysql\_fetch\_array($SQL\_FN)){

?>

<option value="<?php echo $cod\_veiculo; ?>"><?php echo $ftn['placa']; ?></option>

<?php }

}

?>

</select>

</td></tr>

<tr><td>

Data:</td><td><input class="form\_campos" type="text" name="data" size="10"></td><td>

Hora:</td><td><input class="form\_campos" type="text" name="hora" size="3">

</td></tr><tr><td>

Quant. Passageiro:</td><td><input class="form\_campos" type="text" name="quantPassageiro" size="10"></td><td>

Destino:</td><td><input class="form\_campos" type="text" name="destino" size="20">

</td></tr><tr><td>

<input class="form\_botao" type="reset" name="Limpar" value="Limpar">

<input class="form\_botao" onclick="return validar()" type="submit" name="Reservar" value="Reservar" />

</td></tr>

</table>

</form>”

Fragmentos da Pagina devolucao.php.

“<?php

require\_once "includes/config.php";

?>

<body>

<?php

require\_once "includes/cabeca.php"

?>

<div id="principal">

<div id="menu">

<nav>

<ul>

<li><a href="index.php" title="Inicio">Home</a></li>

<li><a href="reserva.php" title="Reserva de veículo">Reserva de veículo</a></li>

<li><a href="verifica.php" title="Verificar Veículo">Verificar Veículo</a></li>

<li><a href="devolucao.php" title="Devolução do veículo" >Devolução do veículo</a></li>

<li><a href="doc/soliReservVeic.pdf" target="\_blank">Doc. de Reserva</a></li>

<li><a href="doc/devolucaoForm.pdf" target="\_blank" >Doc. Devolução</a></li>

</ul>

</nav>

</div>

<div id="centro">

<h1>Devolução do veículo</h1>

<form name="formulario" method="post" action="acoes/devolucao.php" enctype="multipart/form-data">

<table id="tabela" border="0">

<tr>

<td>Cód. da Reserva:</td>

<td><select class="form\_campos" name="codigoReserva">

<?php

$SQL\_E = mysql\_query("SELECT \* FROM status\_veiculo WHERE status='Reservado'");

while($ec = mysql\_fetch\_array($SQL\_E)){

$cod\_veiculo = $ec['cod\_veiculo'];

$SQL\_FN = mysql\_query("SELECT \* FROM veiculo WHERE cod\_veiculo='$cod\_veiculo' ORDER BY cod\_veiculo DESC");

while($ftn = mysql\_fetch\_array($SQL\_FN)){

$SQL\_R = mysql\_query("SELECT \* FROM reserva\_veiculo WHERE cod\_veiculo='$cod\_veiculo'");

while($R = mysql\_fetch\_array($SQL\_R)){

$cod\_res = $R['cod\_reserva'];

}

?>

<option size="10" value="<?php echo $cod\_res; ?>"><?php echo $cod\_res; ?></option>

<?php }}

?>

</select></td>

</tr>

<tr>

<td>Funcionario:</td>

<td><select class="form\_campos" name="codigoFuncionario">

<?php

$SQL\_F = mysql\_query("SELECT \* FROM funcionario WHERE permitir='Sim' ");

while($func = mysql\_fetch\_array($SQL\_F)){

$cod\_func = $func['cod\_funcionario'];

$nome = $func['nome'];

?>

<option size="10" value="<?php echo $cod\_func; ?>"><?php echo $nome; ?></option>

<?php }

?>

</select></td>

<td>Veiculo:</td>

<td><select class="form\_campos" name="codigoVeiculo">

<?php

$SQL\_E = mysql\_query("SELECT \* FROM status\_veiculo WHERE status='Reservado'");

while($ec = mysql\_fetch\_array($SQL\_E)){

$cod\_veiculo = $ec['cod\_veiculo'];

$status = $ec['status'];

$SQL\_FN = mysql\_query("SELECT \* FROM veiculo WHERE cod\_veiculo='$cod\_veiculo' ORDER BY cod\_veiculo DESC LIMIT 2");

while($ftn = mysql\_fetch\_array($SQL\_FN)){

?>

<option size="10" value="<?php echo $cod\_veiculo; ?>"><?php echo $ftn['placa']; ?></option>

<?php }

}

?>

</select></td>

</tr>

<tr>

<td>Km inicial:</td>

<td><input class="form\_campos" type="text" name="kmInicial" size="10"></td>

<td>Km Final:</td>

<td><input class="form\_campos" type="text" name="kmFinal" size="10"></td>

</tr>

<tr>

<td>Data do Retorno:</td>

<td><input class="form\_campos" type="text" name="dataRetorno" size="10">

<td>Hora do Retorno:</td>

<td><input class="form\_campos" type="text" name="horaRetorno" size="10"></td>

</tr>

<tr>

<td><input class="form\_botao" type="reset" name="Limpar" value="Limpar">

<input class="form\_botao" onclick="return validar()" type="submit" name="Reservar" value="Retorno" />

</td>

</tr>

</table>

</form>

”

Fragmento da pagina verificando.php

“

div id="principal">

<div id="menu">

<nav>

<ul>

<li><a href="index.php" title="Inicio">Home</a></li>

<li><a href="reserva.php" title="Reserva de veículo">Reserva de veículo</a></li>

<li><a href="verifica.php" title="Verificar Veículo">Verificar Veículo</a></li>

<li><a href="devolucao.php" title="Devolução do veículo" >Devolução do veículo</a></li>

<li><a href="doc/soliReservVeic.pdf" target="\_blank">Doc. de Reserva</a></li>

<li><a href="doc/devolucaoForm.pdf" target="\_blank">Doc. Devolução</a></li>

</ul>

</nav>

</div>

<div id="centro">

<h1>Verificar Veículo</h1>

<form name="formulario" method="POST" action="verificando.php" enctype="multipart/form-data">

<table id="tabela" border="0">

<tr>

<td>Digite o Código ou Placa do Veículo:</td>

<td><input class="form\_campos" type="text" name="codVeiculo" size="10"></td>

<td><input class="form\_botao" onclick="return validar()" type="submit" name="Reservar" value="Verificar" /></td>

</tr>

</table>

</form>

<br/>

<section id="conteudo">

<?php

$cod\_veiculo = $\_POST["codVeiculo"];

echo "Sua Pesquisa foi por: ".$cod\_veiculo;

$busca = mysql\_query("SELECT \* FROM veiculo WHERE cod\_veiculo='$cod\_veiculo' or placa='$cod\_veiculo'");

if(mysql\_num\_rows($busca) !=0){

while($src = mysql\_fetch\_array($busca)){

?>

<section id='rerifica'>

<h3><?php echo "Placa do Veiculo: ".$src['placa']." - Código: ".$src['cod\_veiculo']; ?></h3>

<?php

$cod = $src['cod\_veiculo'];

$SQL\_S = mysql\_query("SELECT \* FROM status\_veiculo WHERE cod\_veiculo='$cod'");

while($status = mysql\_fetch\_array($SQL\_S)){

$stutusCod = $status['cod\_status'];

$stut = $status['status'];

$string = mb\_convert\_encoding(utf8\_encode($stut), 'UTF-8');

echo "<h4> # ".$string."</h4>";

}

if($stut == 'Reservado'){

$SQL\_R = mysql\_query("SELECT \* FROM reserva\_veiculo WHERE cod\_veiculo='$stutusCod' AND status='Reservado' ORDER BY cod\_veiculo DESC LIMIT 10");

while($res = mysql\_fetch\_array($SQL\_R)){

$func = $res['cod\_funcionario'];

$codR = $res['cod\_reserva'];

$codV = $res['cod\_veiculo'];

echo "<h4> # Numero da Reserva: ".$res['cod\_reserva']."</h4>";

echo "<h4> # Destino: ".$res['destino']."</h4>";

echo "<h4> # Saida em:".$res['data\_saida']." as ".$res['hora\_saida']."</h4>";

echo "<h4> # Passageiros: ".$res['num\_passageiros']."</h4>";

$SQL\_F = mysql\_query("SELECT \* FROM funcionario WHERE cod\_funcionario='$func' ORDER BY cod\_funcionario DESC LIMIT 1");

while($func = mysql\_fetch\_array($SQL\_F)){

echo "<h4> # Reservado por: ".$func['nome']."</h4>"

?>

<form name="formulario" method="POST" action="acoes/cancelar\_reserva.php" enctype="multipart/form-data">

<input class="compo\_branco" type="text" value=" <?php echo $codR; ?>" name="codReserva" size="1">

<input class="compo\_branco" type="text" value=" <?php echo $codV; ?>" name="codVeiculo" size="1">

<input class="form\_botao" onclick="return validar()" type="submit" name="cancelar" value="Cancelar Reserva" />

</form>

<?php } }?>

<?php } ?>

<?php } } else { ?>

<section id='buscar'>

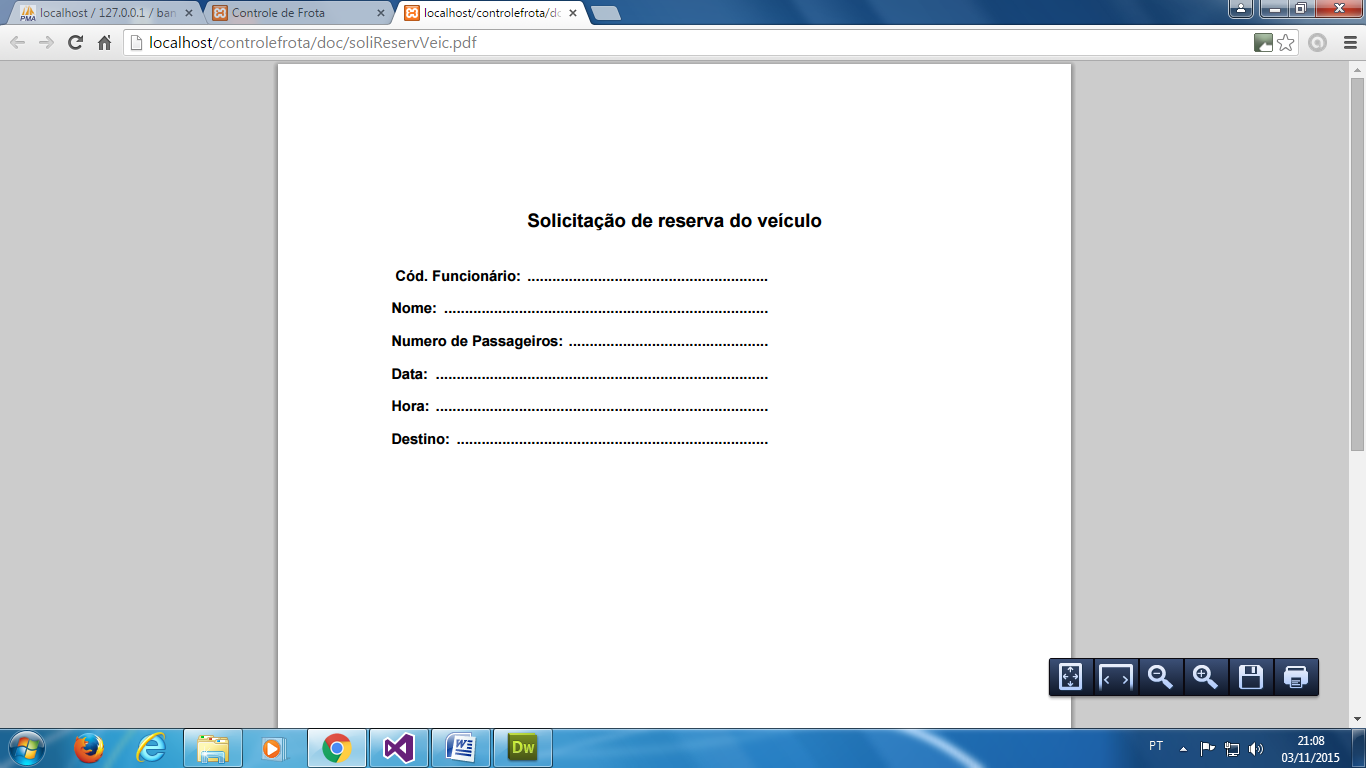
<h1>Desculpe mais não foi encontrado nenhuma Veiculo com esse "Código".</h1>

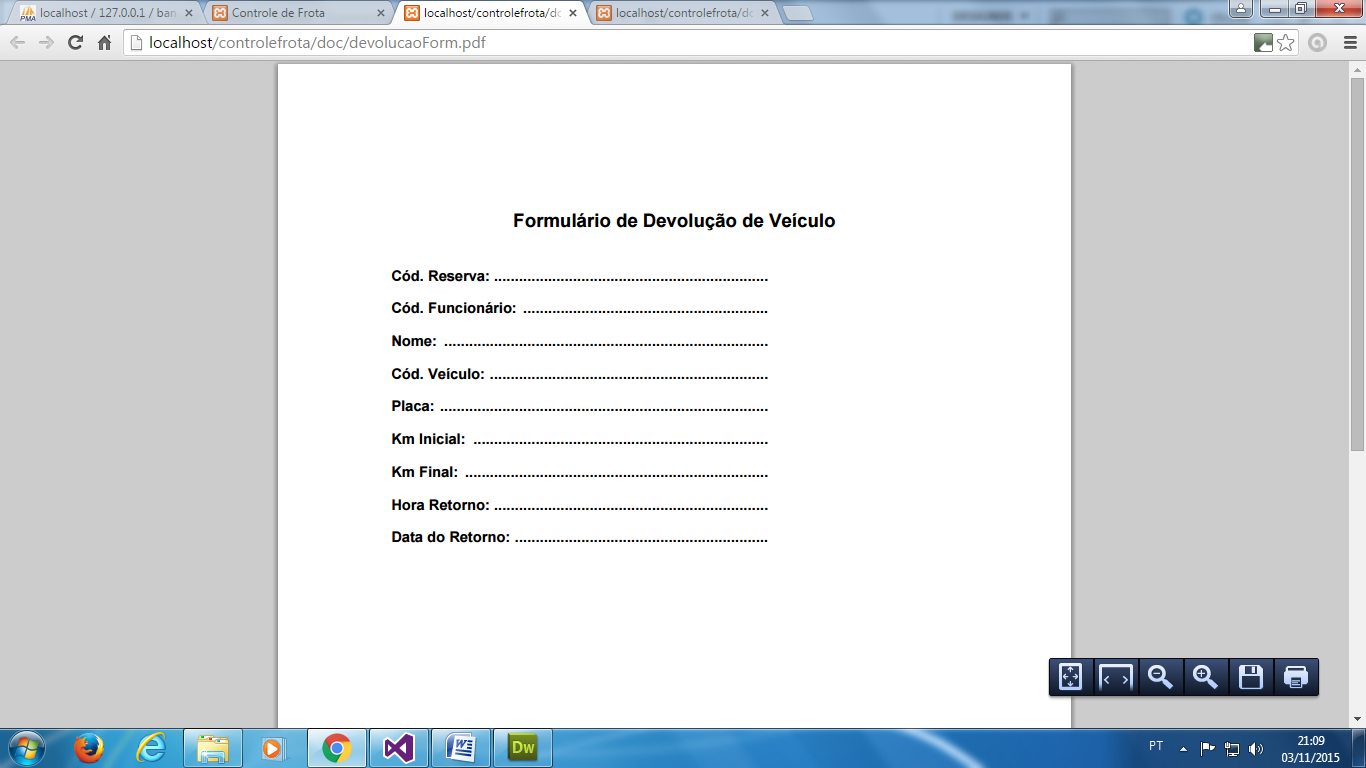
</section>

<?php } ?>”

#### 3.5.3. Geração do Formulários.

Formularios de Devolução e Reserva de veículo.





#### 3.5.4. Geração das Data-base.

Banco de Dados.

-- phpMyAdmin SQL Dump

-- version 4.3.11

-- http://www.phpmyadmin.net

--

-- Host: 127.0.0.1

-- Generation Time: 04-Nov-2015 às 00:13

-- Versão do servidor: 5.6.24

-- PHP Version: 5.6.8

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

SET time\_zone = "+00:00";

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8 \*/;

Database: `banco\_controlefrota`

Estrutura da tabela `devolucao\_veiculo`

CREATE TABLE IF NOT EXISTS `devolucao\_veiculo` (

`cod\_devolucao` int(10) NOT NULL,

`cod\_veiculo` int(10) NOT NULL,

`cod\_funcionario` int(10) NOT NULL,

`cod\_reserva` int(10) NOT NULL,

`data\_retorno` varchar(11) NOT NULL,

`hora\_retorno` varchar(6) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `devolucao\_veiculo`

INSERT INTO `devolucao\_veiculo` (`cod\_devolucao`, `cod\_veiculo`, `cod\_funcionario`, `cod\_reserva`, `data\_retorno`, `hora\_retorno`) VALUES

(1, 1, 10, 3, '28/10/2015', '17:20'),

(2, 2, 14, 2, '29/10/2015', '17:30'),

(3, 1, 10, 4, '30/10/2015', '15:00'),

(4, 2, 10, 5, '28/10/2015', '18:00'),

(5, 1, 10, 7, '30/10/2015', '17:20'),

(6, 1, 10, 11, '30/10/2015', '15:00'),

(7, 1, 14, 19, '03/11/2015', '17:20'),

(8, 1, 14, 19, '03/11/2015', '17:30'),

(9, 5, 11, 20, '04/11/2015', '17:20'),

(10, 5, 11, 21, '03/11/2015', '17:30');

Estrutura da tabela `doc\_funcionario`

CREATE TABLE IF NOT EXISTS `doc\_funcionario` (

`cod\_doc` int(10) NOT NULL,

`cod\_do\_funcionario` int(10) NOT NULL,

`CNH` varchar(15) NOT NULL,

`CPF` varchar(15) NOT NULL,

`num\_id` varchar(15) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=16 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `doc\_funcionario`

INSERT INTO `doc\_funcionario` (`cod\_doc`, `cod\_do\_funcionario`, `CNH`, `CPF`, `num\_id`) VALUES

(10, 10, '0000000000', '1212121212', '1234567890'),

(11, 11, '1111111', '11111111', '1111111'),

(12, 12, '0000000000', '0000000000', '0000000000'),

(13, 13, '22222222222', '22222222222', '22222222222'),

(14, 14, '5555555555', '5555555555', '5555555555'),

(15, 15, '0000000000', '0000000000', '0000000000');

Estrutura da tabela `endereco\_funcionario`

CREATE TABLE IF NOT EXISTS `endereco\_funcionario` (

`cod\_endereco` int(10) NOT NULL,

`cod\_do\_funcionario` int(10) NOT NULL,

`cidade` varchar(50) NOT NULL,

`rua` varchar(50) NOT NULL,

`bairro` varchar(50) NOT NULL,

`referecia` varchar(50) NOT NULL,

`numero` varchar(10) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=13 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `endereco\_funcionario`

INSERT INTO `endereco\_funcionario` (`cod\_endereco`, `cod\_do\_funcionario`, `cidade`, `rua`, `bairro`, `referecia`, `numero`) VALUES

(7, 10, 'Garibaldi', 'Gaspar', 'Ferro', 'Ap01', '0151'),

(8, 11, 'Caxias', 'Rua', 'Centro', 'Ap305', '325'),

(9, 12, 'Caxias', 'Rua', 'Centro', 'Ap', '105'),

(10, 13, 'Caxias', 'Rua55', 'Bairro1', 'Casa', '123'),

(11, 14, 'Caxias', 'Rua36', 'Centro', 'Casa', '123'),

(12, 15, 'Caxias', 'Rua01', 'Centro', 'Ap01', '1234');

Estrutura da tabela `funcionario`

CREATE TABLE IF NOT EXISTS `funcionario` (

`cod\_funcionario` int(10) NOT NULL,

`nome` varchar(50) NOT NULL,

`data\_nasc` varchar(11) NOT NULL,

`fone` varchar(15) NOT NULL,

`permitir` varchar(10) NOT NULL,

`categoria\_cnh` varchar(10) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=16 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `funcionario`

INSERT INTO `funcionario` (`cod\_funcionario`, `nome`, `data\_nasc`, `fone`, `permitir`, `categoria\_cnh`) VALUES

(10, 'Douglas', '12/05/1989', '54,555555555', 'Sim', 'C'),

(11, 'Maria', '05/06/1985', '542222-2222', 'Sim', 'C'),

(12, 'Dany', '05/05/1989', '(54)0000-0000', 'Não', 'B'),

(13, 'Mario', '25/02/1988', '555555555555', 'Não', 'B'),

(14, 'Luis', '14/08/1984', '5488888888', 'Sim', 'B'),

(15, 'Lucas', '15/07/1985', '54-96484746', 'Sim', 'C');

Estrutura da tabela `km\_veiculos`

CREATE TABLE IF NOT EXISTS `km\_veiculos` (

`cod\_km` int(10) NOT NULL,

`cod\_veiculo` int(10) NOT NULL,

`cod\_devolucao` int(10) NOT NULL,

`km\_inicial` varchar(20) NOT NULL,

`km\_final` varchar(20) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=10 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `km\_veiculos`

INSERT INTO `km\_veiculos` (`cod\_km`, `cod\_veiculo`, `cod\_devolucao`, `km\_inicial`, `km\_final`) VALUES

(1, 1, 1, '15800', '16000'),

(2, 1, 3, '18200', '18250'),

(3, 2, 4, '10500', '10700'),

(4, 1, 5, '10700', '10750'),

(5, 1, 6, '18500', '18700'),

(6, 1, 7, '18700', '18800'),

(7, 1, 8, '18800', '18900'),

(8, 5, 9, '18200', '18250'),

(9, 5, 10, '18200', '18250');

Estrutura da tabela `reserva\_veiculo`

CREATE TABLE IF NOT EXISTS `reserva\_veiculo` (

`cod\_reserva` int(10) NOT NULL,

`cod\_funcionario` int(10) NOT NULL,

`cod\_veiculo` int(10) NOT NULL,

`num\_passageiros` int(60) NOT NULL,

`data\_saida` varchar(11) NOT NULL,

`hora\_saida` varchar(6) NOT NULL,

`destino` varchar(50) NOT NULL,

`status` varchar(50) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=23 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `reserva\_veiculo`

INSERT INTO `reserva\_veiculo` (`cod\_reserva`, `cod\_funcionario`, `cod\_veiculo`, `num\_passageiros`, `data\_saida`, `hora\_saida`, `destino`, `status`) VALUES

(2, 13, 2, 5, '29/10/2015', '07:30', 'Bento GonÃ§alves - RS', 'Devolvido'),

(4, 10, 1, 4, '30/10/2015', '07:30', 'Bento GonÃ§alves - RS', 'Devolvido'),

(5, 10, 2, 4, '28/10/2015', '12:30', 'Porto Alegre - RS', 'Devolvido'),

(7, 10, 1, 4, '30/10/2015', '12:30', 'Bento GonÃ§alves - RS', 'Devolvido'),

(8, 10, 1, 4, 'Cancelada', 'Cancel', 'Cancelado', 'Cancelado'),

(9, 10, 1, 4, '30/10/2015', '12:30', 'Bento GonÃ§alves - RS', 'Devolvido'),

(10, 10, 2, 4, 'Cancelada', 'Cancel', 'Cancelado', 'Cancelado'),

(11, 10, 2, 4, '31/10/2015', '07:30', 'Porto Alegre - RS', 'Devolvido'),

(12, 10, 1, 10, 'Cancelada', 'Cancel', 'Cancelado', 'Cancelado'),

(13, 10, 2, 10, 'Cancelada', 'Cancel', 'Cancelado', 'Cancelado'),

(14, 10, 3, 10, 'Cancelada', 'Cancel', 'Cancelado', 'Cancelado'),

(15, 10, 2, 4, 'Cancelada', 'Cancel', 'Cancelado', 'Cancelado'),

(16, 10, 2, 4, '02/11/2015', '8:30', 'Gramado', 'Cancelado'),

(17, 10, 3, 15, '31/10/2015', '12:30', 'Caxias do Sul - RS', 'Cancelado'),

(18, 10, 3, 10, '01/11/2015', '07:30', 'Porto Alegre - RS', 'Cancelado'),

(19, 14, 1, 4, '03/11/2015', '07:30', 'Caxias do Sul - RS', 'Devolvido'),

(20, 11, 5, 4, '04/11/2015', '07:30', 'Bento GonÃ§alves - RS', 'Devolvido'),

(21, 11, 5, 4, '03/11/2015', '07:30', 'Porto Alegre - RS', 'Devolvido'),

(22, 10, 2, 4, '04/11/2015', '07:30', 'Bento GonÃ§alves - RS', 'Reservado');

Estrutura da tabela `status\_veiculo`

CREATE TABLE IF NOT EXISTS `status\_veiculo` (

`cod\_status` int(10) NOT NULL,

`cod\_veiculo` int(10) NOT NULL,

`status` varchar(50) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `status\_veiculo`

INSERT INTO `status\_veiculo` (`cod\_status`, `cod\_veiculo`, `status`) VALUES

(1, 1, 'Disponivel'),

(2, 2, 'Reservado'),

(3, 3, 'Disponivel'),

(4, 4, 'Desativado'),

(5, 5, 'Disponivel'),

(6, 6, 'Desativado');

Estrutura da tabela `veiculo`

CREATE TABLE IF NOT EXISTS `veiculo` (

`cod\_veiculo` int(10) NOT NULL,

`placa` varchar(10) NOT NULL,

`marca\_montadora` varchar(10) NOT NULL,

`modelo` varchar(10) NOT NULL,

`ano` varchar(5) NOT NULL,

`cor` varchar(10) NOT NULL,

`num\_portas` varchar(5) NOT NULL,

`num\_passageiros` int(60) NOT NULL,

`tipo\_combustivel` varchar(15) NOT NULL,

`opcionais` varchar(200) NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=latin1;

Extraindo dados da tabela `veiculo`

INSERT INTO `veiculo` (`cod\_veiculo`, `placa`, `marca\_montadora`, `modelo`, `ano`, `cor`, `num\_portas`, `num\_passageiros`, `tipo\_combustivel`, `opcionais`) VALUES

(1, 'IKP0854', 'Volkswagen', 'Gol', '2008', 'Branco', '4', 5, 'Gasolina', 'Ar Quente'),

(2, 'INH8554', 'volkswagen', 'Gol', '2015', 'Branco', '4', 5, 'Gasolina', 'Ar quente'),

(3, 'JKM5240', 'Renault', 'Van', '2010', 'Branco', '3', 16, 'Diesel', 'Vidro Eletrico, trava Eletrica, sistema de Som, Porta automatica'),

(4, 'DGI1289', 'Fiat', 'Uno', '2013', 'Branco', '4', 5, 'Gaslina', 'Ar quente'),

(5, 'IJD0254', 'volkswagen', 'Gol', '2014', 'Branco', '4', 5, 'Gasolina', 'Ar Quente'),

(6, 'IKH0515', 'Renault', 'Vam', '2013', 'Branco', '3', 16, 'Disel', 'Ar quente, porta automatica');

Indexes for dumped tables

Indexes for table `devolucao\_veiculo`

ALTER TABLE `devolucao\_veiculo`

ADD PRIMARY KEY (`cod\_devolucao`);

Indexes for table `doc\_funcionario`

ALTER TABLE `doc\_funcionario`

ADD PRIMARY KEY (`cod\_doc`), ADD KEY `fk\_funcionario` (`cod\_do\_funcionario`);

Indexes for table `endereco\_funcionario`

ALTER TABLE `endereco\_funcionario`

ADD PRIMARY KEY (`cod\_endereco`), ADD KEY `fk\_doc\_funcionario` (`cod\_do\_funcionario`);

Indexes for table `funcionario`

ALTER TABLE `funcionario`

ADD PRIMARY KEY (`cod\_funcionario`);

Indexes for table `km\_veiculos`

ALTER TABLE `km\_veiculos`

ADD PRIMARY KEY (`cod\_km`);

Indexes for table `reserva\_veiculo`

ALTER TABLE `reserva\_veiculo`

ADD PRIMARY KEY (`cod\_reserva`);

Indexes for table `status\_veiculo`

ALTER TABLE `status\_veiculo`

ADD PRIMARY KEY (`cod\_status`);

Indexes for table `veiculo`

ALTER TABLE `veiculo`

ADD PRIMARY KEY (`cod\_veiculo`);

AUTO\_INCREMENT for dumped tables

AUTO\_INCREMENT for table `devolucao\_veiculo`

ALTER TABLE `devolucao\_veiculo`

MODIFY `cod\_devolucao` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=11;

AUTO\_INCREMENT for table `doc\_funcionario`

ALTER TABLE `doc\_funcionario`

MODIFY `cod\_doc` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=16;

AUTO\_INCREMENT for table `endereco\_funcionario`

ALTER TABLE `endereco\_funcionario`

MODIFY `cod\_endereco` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=13;

ALTER TABLE `funcionario`

MODIFY `cod\_funcionario` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=16;

AUTO\_INCREMENT for table `km\_veiculos`

ALTER TABLE `km\_veiculos`

MODIFY `cod\_km` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=10;

AUTO\_INCREMENT for table `reserva\_veiculo`

ALTER TABLE `reserva\_veiculo`

MODIFY `cod\_reserva` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=23;

AUTO\_INCREMENT for table `status\_veiculo`

ALTER TABLE `status\_veiculo`

MODIFY `cod\_status` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=7;

AUTO\_INCREMENT for table `veiculo`

ALTER TABLE `veiculo`

MODIFY `cod\_veiculo` int(10) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=7;

Constraints for dumped tables

Limitadores para a tabela `doc\_funcionario`

ALTER TABLE `doc\_funcionario`

ADD CONSTRAINT `fk\_funcionario` FOREIGN KEY (`cod\_do\_funcionario`) REFERENCES `funcionario` (`cod\_funcionario`);

Limitadores para a tabela `endereco\_funcionario`

ALTER TABLE `endereco\_funcionario`

ADD CONSTRAINT `fk\_doc\_funcionario` FOREIGN KEY (`cod\_do\_funcionario`) REFERENCES `doc\_funcionario` (`cod\_do\_funcionario`);

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

#### 4. CONCLUSÃO

Este trabalho expõe a importância das tecnologias o quanto devemos nos atualizar, proporcionando o conhecimento mais aprofundado em Analise e Desenvolvimento de Sistemas.

Com a Programação Orientada a Objetos, aprimoramos nossos conhecimentos e objetivamos mais o foco no mundo real assim podemos simplificar todo o projeto onde todos pode dar continuidade no projeto.

Banco de Dados foca muito no uso do BRModelo que nos orienta graficamente como funciona o gerenciamento do nosso sistema.

Com a Programação WEB, usamos e trabalhamos com o PHP uma tecnologia de programação WEB vinculada ao Banco de Dados onde é possível desenvolver e aplicar muitas rotinas para sites dinâmicos.

Também usamos a IDE C# para desenvolver um sistema de cadastro onde neste trabalho aprimorando o conhecimento com a Linguagem e aplicando de forma pratica no cadastro de funcionários.

Toda a pratica com este trabalho mostra que podemos aprender muito com ADS e usar este conhecimento para desenvolver sistemas a aplica-los em qualquer lugar do mercado.

#### 5. REFERÊNCIA

**NISHIMURA,** Roberto Yukio. **Banco de Dados II,** São Paulo: Editora Pearson Education do Brasil, 2009.

**TANAKA,** Simone Sawasaki. **Análise de Sistemas III.** São Paulo: Editora Pearson Education do Brasil, 2013.

**SOLER**, Luciano e **Moraes**, Everson Matias de. **Desenvolvimento de Aplicação Web.** São Paulo: Editora Pearson Education do Brasil, 2010.

**SILVA**, Flávio de Almeida e, **Desenvolvimento Orientado a Objetos I**. São Paulo: Pearson Prentice Hall, 2009.

DEVMEDIA, Modelo Entidade Relacionamento. Disponível em:

**<http://www.devmedia.com.br/modelo-entidade-relacionamento-mer-e-diagrama-entidade-relacionamento-der/14332>** Acessado em: set/2015

DEVMEDIA, O que é HTML5, Disponível em**: <http://www.devmedia.com.br/o-que-e-o-html5/25820>** acessado em: out/2015

HLERA, Tecnologias. Disponível em: **<http://hleranetworks.com.br/tecnologias-utilizadas.htm>** acessado em: out/2015

HTML5, As novas Tecnologias. Disponível em: **<http://pt.slideshare.net/talitapagani/html5-e-as-novas-tecnologias-de-desenvolvimento-web>** acessado em: out/2015

ITNERANTE, Desmistificando as formas normais, disponível em: **<http://www.itnerante.com.br/profiles/blogs/desmistificando-as-formas-normais>** acessado em: nov/2015

SIGNIFICADOS, HTML. Disponível e: **<http://www.significados.com.br/html/>** acessado em: out/2015

Vídeos:

**< https://www.youtube.com/watch?v=F7KzJ7e6EAc>**

**<https://www.youtube.com/watch?v=uNzd9y5YmSc&list=PL7195B3630F7FC4F>**

**<https://www.youtube.com/watch?v=fvb3e1qcRTs>**