**App\_Start**

**Class - BundleConfig**

using System.Web;

using System.Web.Optimization;

namespace HelpDesk

{

public class BundleConfig

{

// Para obter mais informações sobre o agrupamento, visite https://go.microsoft.com/fwlink/?LinkId=301862

public static void RegisterBundles(BundleCollection bundles)

{

bundles.Add(new ScriptBundle("~/bundles/jquery").Include(

"~/Scripts/jquery-{version}.js"));

// Use a versão em desenvolvimento do Modernizr para desenvolver e aprender com ela. Após isso, quando você estiver

// pronto para a produção, utilize a ferramenta de build em https://modernizr.com para escolher somente os testes que precisa.

bundles.Add(new ScriptBundle("~/bundles/modernizr").Include(

"~/Scripts/modernizr-\*"));

bundles.Add(new ScriptBundle("~/bundles/bootstrap").Include(

"~/Scripts/bootstrap.js"));

bundles.Add(new StyleBundle("~/Content/css").Include(

"~/Content/bootstrap.css",

"~/Content/site.css"));

}

}

}

**Class - FilterConfig**

using System.Web;

using System.Web.Mvc;

namespace HelpDesk

{

public class FilterConfig

{

public static void RegisterGlobalFilters(GlobalFilterCollection filters)

{

filters.Add(new HandleErrorAttribute());

}

}

}

**Class - RouteConfig**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using System.Web.Routing;

namespace HelpDesk

{

public class RouteConfig

{

public static void RegisterRoutes(RouteCollection routes)

{

routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

routes.MapRoute(

name: "Default",

url: "{controller}/{action}/{id}",

defaults: new { controller = "Login", action = "Index", id = UrlParameter.Optional }

);

}

}

}

**Class - WebApiConfig**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web.Http;

namespace HelpDesk

{

public static class WebApiConfig

{

public static void Register(HttpConfiguration config)

{

// Serviços e configuração da API da Web

// Rotas da API da Web

config.MapHttpAttributeRoutes();

config.Routes.MapHttpRoute(

name: "DefaultApi",

routeTemplate: "api/{controller}/{id}",

defaults: new { id = RouteParameter.Optional }

);

}

}

}

**Controllers**

**Class - ChamadosController**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web;

using System.Web.Http;

using System.Web.Mvc;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class ChamadosController : ApiController

{

public static Context db = new Context();

// GET api/Chamados

public IEnumerable<Chamados> Get()

{

return db.Chamados.ToList();

}

// GET api/Chamados/5

public Chamados Get(string id)

{

return db.Chamados.ToList().Where(chamado => chamado.Cod\_Chamado == id).First();

}

// POST api/Chamados

public HttpResponseMessage Post([FromBody]Chamados value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

var chamados = db.Chamados.ToList();

value.Cod\_Chamado = (chamados.Count() + 1).ToString();

value.Data = DateTime.Now;

db.Chamados.Add(value);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// PUT api/Chamados/5

public HttpResponseMessage Put(int id, [FromBody]Chamados value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Chamados.Add(value);

db.Entry(value).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// DELETE api/Chamados/5

public HttpResponseMessage Delete(string id)

{

var chamado = db.Chamados.ToList().Where(c => c.Cod\_Chamado == id).First();

db.Chamados.Remove(chamado);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

}

}

**Class - ClienteController**

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class ClienteController : ApiController

{

public static Context db = new Context();

// GET api/values

public IEnumerable<Cliente> Get()

{

return db.Cliente.ToList();

}

// GET api/values/5

public Cliente Get(string id)

{

return db.Cliente.ToList().Where(Cliente => Cliente.Cod\_Cliente == id).First();

}

// POST api/values

public HttpResponseMessage Post([FromBody]Cliente value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Cliente.Add(value);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// PUT api/values/5

public HttpResponseMessage Put(int id, [FromBody]Cliente value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Cliente.Add(value);

db.Entry(value).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// DELETE api/values/5

public HttpResponseMessage Delete(string id)

{

var chamado = db.Cliente.ToList().Where(c => c.Cod\_Cliente == id).First();

db.Cliente.Remove(chamado);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

}

}

**Class - DashboardController**

using System.Linq;

using System.Web.Mvc;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class DashboardController : Controller

{

private static Context db = new Context();

public ActionResult Index()

{

string Cod\_usuario = (string)@Session["Login"];

ViewBag.Cod\_usuario = Cod\_usuario;

if (Cod\_usuario == null)

{

return View();

}

else

{

Usuario usuario;

Cliente cliente = null;

cliente = db.Cliente.ToList().Where(c => c.ID\_usuario == Cod\_usuario).FirstOrDefault();

if (cliente != null)

{

usuario = db.Usuario.ToList().Where(u => u.ID\_usuario == cliente.ID\_usuario).FirstOrDefault();

ViewBag.Chamados = new Chamados().GetListByUsuer(Cod\_usuario.ToString());

ViewBag.IsCliente = true;

}

else

{

Funcionario funcionario = null;

funcionario = db.Funcionario.ToList().Where(f => f.ID\_usuario == Cod\_usuario).FirstOrDefault();

if (funcionario != null)

{

usuario = db.Usuario.ToList().Where(u => u.ID\_usuario == funcionario.ID\_usuario).First();

ViewBag.Chamados = new Chamados().GetList();

ViewBag.IsCliente = false;

}

}

ViewBag.Title = "Dashboard";

ViewBag.Produtos = new Produto().GetProdutos(Cod\_usuario.ToString());

}

return View();

}

}

}

**Class - FuncionarioController**

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class FuncionarioController : ApiController

{

public static Context db = new Context();

// GET api/values

public IEnumerable<Funcionario> Get()

{

return db.Funcionario.ToList();

}

// GET api/values/5

public Funcionario Get(string id)

{

return db.Funcionario.ToList().Where(funcionario => funcionario.Cod\_Funcionario == id).First();

}

// POST api/values

public HttpResponseMessage Post([FromBody]Funcionario value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Funcionario.Add(value);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// PUT api/values/5

public HttpResponseMessage Put(int id, [FromBody]Funcionario value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Funcionario.Add(value);

db.Entry(value).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// DELETE api/values/5

public HttpResponseMessage Delete(string id)

{

var chamado = db.Funcionario.ToList().Where(c => c.Cod\_Funcionario == id).First();

db.Funcionario.Remove(chamado);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

}

}

**Class - LoginController**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Web.Http;

using System.Web.Mvc;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class LoginController : Controller

{

public ActionResult Index([FromBody]string email, string senha)

{

ViewBag.Title = "Home Page";

if (email != null && senha != null)

{

var result = Usuario.Login(email, senha);

if (result != null)

{

Session["Login"] = result;

Response.Redirect("~/Dashboard/Index");

return null;

}

}

return View();

}

}

}

**Class - ProdutoController**

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class ProdutoController : ApiController

{

public static Context db = new Context();

// GET api/values

public IEnumerable<Produto> Get()

{

return db.Produto.ToList();

}

// GET api/values/5

public Produto Get(string id)

{

return db.Produto.ToList().Where(produto => produto.Cod\_Produto == id).First();

}

// POST api/values

public HttpResponseMessage Post([FromBody]Produto value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Produto.Add(value);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// PUT api/values/5

public HttpResponseMessage Put(int id, [FromBody]Produto value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Produto.Add(value);

db.Entry(value).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// DELETE api/values/5

public HttpResponseMessage Delete(string id)

{

var chamado = db.Produto.ToList().Where(c => c.Cod\_Produto == id).First();

db.Produto.Remove(chamado);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

}

}

**Class - TipoAtendimentoController**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class TipoAtendimentoController : ApiController

{

public static Context db = new Context();

// GET api/values

public IEnumerable<TipoAtendimento> Get()

{

return db.TipoAtendimento.ToList();

}

// GET api/values/5

public TipoAtendimento Get(string id)

{

return db.TipoAtendimento.ToList().Where(tipoAtendimento => tipoAtendimento.Cod\_Atendimento == id).First();

}

// POST api/values

public TipoAtendimento Post([FromBody]TipoAtendimento value)

{

if (value == null)

return null;

var tipo = db.TipoAtendimento.ToList();

value.Cod\_Atendimento = (tipo.Count() + 1).ToString();

value.Data = DateTime.Now;

db.TipoAtendimento.Add(value);

db.SaveChanges();

return value;

}

// PUT api/values/5

public HttpResponseMessage Put(int id, [FromBody]TipoAtendimento value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.TipoAtendimento.Add(value);

db.Entry(value).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// DELETE api/values/5

public HttpResponseMessage Delete(string id)

{

var chamado = db.TipoAtendimento.ToList().Where(c => c.Cod\_Atendimento == id).First();

db.TipoAtendimento.Remove(chamado);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

}

}

**Class - UploadFileController**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web;

using System.Web.Http;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class UploadFileController : ApiController

{

public HttpResponseMessage Post()

{

var file = HttpContext.Current.Request.Files.Count > 0 ?

HttpContext.Current.Request.Files[0] : null;

if (file != null && file.ContentLength > 0)

{

var fileName = Path.GetFileName(file.FileName);

var path = Path.Combine(

HttpContext.Current.Server.MapPath("~/UploadedFiles"),

DateTime.Now.ToString("dd-MM-yyyy") + fileName

);

file.SaveAs(path);

return Request.CreateResponse(HttpStatusCode.OK);

}

return Request.CreateResponse(HttpStatusCode.InternalServerError);

}

}

}

**Class - UsuarioController**

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Net.Http;

using System.Web.Http;

using HelpDesk.Models;

namespace HelpDesk.Controllers

{

public class UsuarioController : ApiController

{

public static Context db = new Context();

// GET api/values

public IEnumerable<Usuario> Get()

{

var t = db.Usuario.ToList(); ;

return t;

}

// GET api/values/5

public Usuario Get(string id)

{

return db.Usuario.ToList().Where(usuario => usuario.ID\_usuario == id).First();

}

// POST api/values

public HttpResponseMessage Post([FromBody]Usuario value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Usuario.Add(value);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// PUT api/values/5

public HttpResponseMessage Put(int id, [FromBody]Usuario value)

{

if (value == null)

return Request.CreateResponse(HttpStatusCode.BadRequest);

db.Usuario.Add(value);

db.Entry(value).State = System.Data.Entity.EntityState.Modified;

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

// DELETE api/values/5

public HttpResponseMessage Delete(string id)

{

var usuario = db.Usuario.ToList().Where(u => u.ID\_usuario == id).First();

db.Usuario.Remove(usuario);

db.SaveChanges();

return Request.CreateResponse(HttpStatusCode.OK);

}

}

}

**Models**

**Class - Chamados**

namespace HelpDesk.Models

{

using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.Serialization;

public partial class Chamados

{

[System.ComponentModel.DataAnnotations.Key]

public string Cod\_Chamado { get; set; }

public string Desc\_Chamado { get; set; }

public Nullable<System.DateTime> Data { get; set; }

public string Cod\_Produto { get; set; }

public string Cod\_Atendimento { get; set; }

public virtual TipoAtendimento TipoAtendimento { get; set; }

public virtual Produto Produto { get; set; }

private static Context db = new Context();

public List<Chamados> GetListByUsuer(string cod)

{

try

{

List<Chamados> list = new List<Chamados>();

List<Chamados> chamados = db.Chamados.ToList();

foreach (Chamados c in chamados)

{

c.Produto = db.Produto.ToList().Where(p => p.Cod\_Produto == c.Cod\_Produto).ToList<Produto>().FirstOrDefault();

c.TipoAtendimento = db.TipoAtendimento.ToList().Where(tipoAtendimento => tipoAtendimento.Cod\_Atendimento == c.Cod\_Atendimento).First();

if (c.Produto.Cod\_Cliente == cod)

list.Add(c);

}

return list.OrderByDescending(c => c.Data).ToList(); ;

}

catch

{

return null;

}

}

public List<Chamados> GetList()

{

List<Chamados> chamados = db.Chamados.ToList().OrderByDescending(c => c.Data).ToList();

return chamados;

}

}

}

**Class - Cliente**

namespace HelpDesk.Models

{

using System;

using System.Collections.Generic;

public partial class Cliente

{

[System.ComponentModel.DataAnnotations.Key]

public string Cod\_Cliente { get; set; }

public string Nome { get; set; }

public string Razao\_Social { get; set; }

public string CPF { get; set; }

public string CNPJ { get; set; }

public string Email\_Contato { get; set; }

public string End\_Completo { get; set; }

public string Telefone { get; set; }

public char Stat\_Cliente { get; set; }

public string ID\_usuario { get; set; }

public virtual Usuario Usuario { get; set; }

public virtual ICollection<Produto> Produto { get; set; }

}

}

**Class - Context**

namespace HelpDesk.Models

{

using System;

using System.Data.Entity;

using System.ComponentModel.DataAnnotations.Schema;

using System.Linq;

using System.Data.Entity.ModelConfiguration.Conventions;

/\* Classe context é responsavel por fazer as interaçoes entre os modelos e o banco de dados

\* sendo necessario apenas mapealos utilizando o objeto DbSete.

\* BdSete é umma classe que implementa o entity Framework

\*/

public class Context : DbContext

{

public DbSet<Chamados> Chamados { get; set; }

public DbSet<Cliente> Cliente { get; set; }

public DbSet<Funcionario> Funcionario { get; set; }

public DbSet<Produto> Produto { get; set; }

public DbSet<TipoAtendimento> TipoAtendimento { get; set; }

public DbSet<Usuario> Usuario { get; set; }

protected override void OnModelCreating(DbModelBuilder modelBuilder)

{

modelBuilder.Conventions.Remove<PluralizingTableNameConvention>();

base.OnModelCreating(modelBuilder);

}

public Context()

{

this.Configuration.ProxyCreationEnabled = false;

}

}

}

**Class - Funcionario**

namespace HelpDesk.Models

{

using System;

using System.Collections.Generic;

public partial class Funcionario

{

[System.ComponentModel.DataAnnotations.Key]

public string Cod\_Funcionario { get; set; }

public string Nome\_Completo { get; set; }

public string Nome\_Tratamento { get; set; }

public string CPF { get; set; }

public string End\_Completo { get; set; }

public string Telefone { get; set; }

public string Email\_Contato { get; set; }

public char Stat\_Funcionario { get; set; }

public string ID\_usuario { get; set; }

public virtual Usuario Usuario { get; set; }

}

}

**Class - Produto**

//------------------------------------------------------------------------------

// <auto-generated>

// This code was generated from a template.

//

// Manual changes to this file may cause unexpected behavior in your application.

// Manual changes to this file will be overwritten if the code is regenerated.

// </auto-generated>

//------------------------------------------------------------------------------

namespace HelpDesk.Models

{

using System;

using System.Collections;

using System.Collections.Generic;

using System.Linq;

public partial class Produto

{

[System.ComponentModel.DataAnnotations.Key]

public string Cod\_Produto { get; set; }

public string Desc\_Produto { get; set; }

public string Cod\_Cliente { get; set; }

private static Context db = new Context();

public List<Produto> GetProdutos(string codCliente)

{

return db.Produto.ToList().Where(p => p.Cod\_Cliente == codCliente).ToList<Produto>();

}

}

}

**Class - TipoAtendimento**

namespace HelpDesk.Models

{

using System;

using System.Collections.Generic;

public partial class TipoAtendimento

{

[System.ComponentModel.DataAnnotations.Key]

public string Cod\_Atendimento { get; set; }

public string Descricao { get; set; }

public Nullable<System.DateTime> Data { get; set; }

public string Prioridade { get; set; }

}

}

**Class - Usuario**

namespace HelpDesk.Models

{

using System;

using System.Collections.Generic;

using System.Linq;

public partial class Usuario

{

[System.ComponentModel.DataAnnotations.Key] // System informa que é a chave primaria, porque nao foi utiizada nomenclatura padrao no banco que seria ID

public string ID\_usuario { get; set; }

public char Acesso { get; set; }

public string Senha { get; set; }

private static Context db = new Context();

public static string Login(String email, String senha)

{

Usuario usuario = null;

try {

Cliente cliente = null;

cliente = db.Cliente.ToList().Where(c => c.Email\_Contato == email).FirstOrDefault();

if (cliente != null)

{

usuario = db.Usuario.ToList().Where(u => u.ID\_usuario == cliente.ID\_usuario).FirstOrDefault();

if (usuario.Senha == senha)

return usuario.ID\_usuario;

}

Funcionario funcionario = null;

funcionario = db.Funcionario.ToList().Where(f => f.Email\_Contato == email).FirstOrDefault();

if (funcionario != null)

{

usuario = db.Usuario.ToList().Where(u => u.ID\_usuario == funcionario.ID\_usuario).FirstOrDefault();

return usuario.ID\_usuario;

}

return null;

} catch (Exception ex)

{

return null;

}

}

}

}