

UNIÃO DAS ESCOLAS SUPERIORES DE RONDÔNIA – UNIRON
CURSO DE SISTEMAS DE INFORMAÇÃO

PROJETO INTEGRADOR V (SISTEMAS RESPONSIVOS)

GRUPO

EDUARDO HENRIQUE
LUIZ FELIPE
PAULO SÉRGIO
YAN ROQUE

PORTO VELHO/RO

1. INTRODUÇÃO

Atualmente as pessoas no mundo têm uma vida muito atarefada, por vezes necessitando de uma agenda para melhor organizar seus compromissos e tarefas. Uma maneira fácil e rápida para lembrar dos compromissos e não esquecê-los. Buscar uma maneira melhor para organizar seus afazeres torna seu dia mais proveitoso, podendo utilizar cada minuto, da forma mais proveitosa.

2. ESCOPO

1. Justificativa do Projeto

As Pessoas possuem hoje muitos afazeres, e a falta de organização impacta nas suas agendas de forma que atrasa sua planos e afeta sua vida, tanto corporativa e pessoal.

Observando esses problemas, acreditamos que esse projeto irá auxiliar no suporte para organização pessoal e melhorar o desempenho de profissionais de todas as áreas que utilizarem o produto.

2.2 Finalidade do Projeto

Implantar o produto no meio corporativo para reduzir a falta de organização das tarefas.

2.3 Objetivo(s) do Projeto

- Identificar as pendências do cliente na parte da organização da agenda.
- Criar documentação e formas de controle para o projeto.

2.4 Descrição do Produto

O sistema será online, cada colaborador terá sua própria trilha para desenvolver suas atividades dentro do sistema, focado na melhoria pessoal e competências. O colaborador só terá acesso a uma visão, que é o seu layout.

3. DESCRIÇÃO DOS USUÁRIOS

O projeto é indicado para qualquer usuário, tanto para o uso corporativo ou pessoal. Como já dito o intuito do projeto é auxiliar na agenda e tarefas, não tendo uma especificação para o usuário ideal. Mas o usuário que for utilizar tem que ter habituado com tipo de produto e tipos de frameworks que servem para gerência.

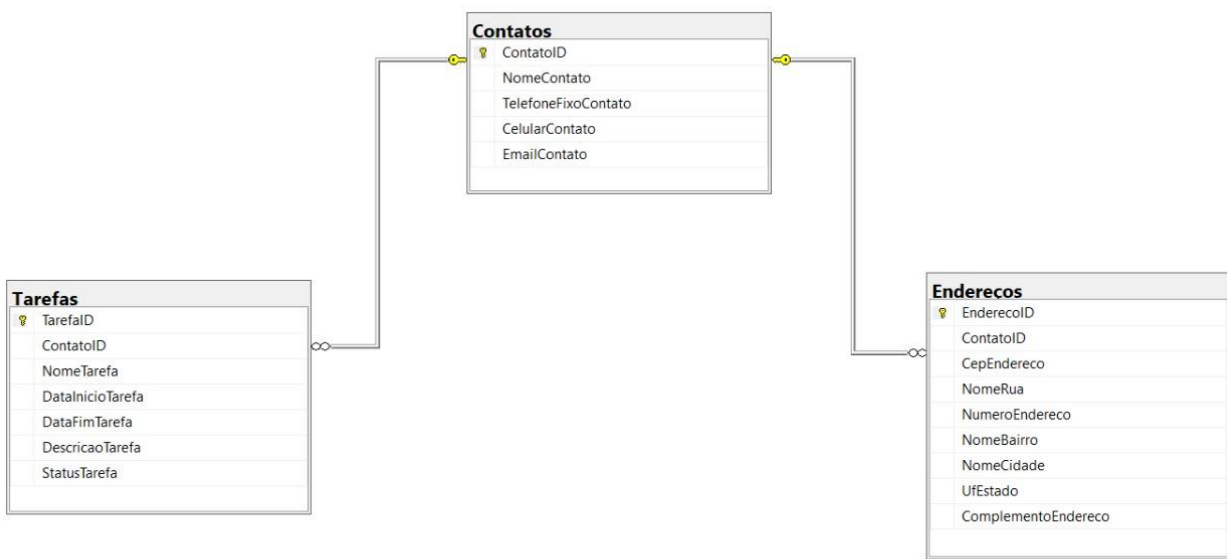
4. REQUISITOS FUNCIONAIS

- O usuário poderá criar diversas listas e tarefas personalizadas.
- O usuário poderá modificar, excluir e ocultar listas.
- O usuário poderá adicionar diversas tarefas em uma lista.
- O usuário poderá modificar, excluir e marcar como concluída as tarefas.
- O sistema armazenará as listas em um banco de dados.

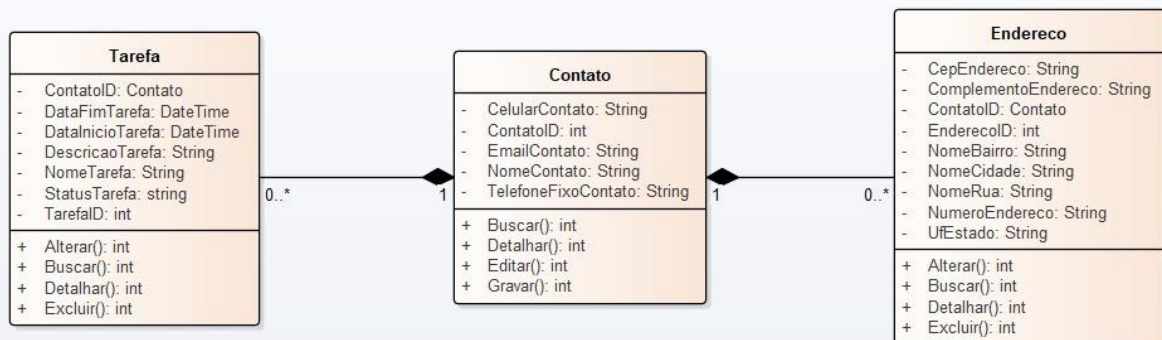
5. REQUISITOS NÃO FUNCIONAIS

- Linguagem HTML, CSS e JavaScript
- Framework Bootstrap
- Microsoft SQL Server
- EntityFramework

6. DIAGRAMA DE ENTIDADE-RELACIONAMENTO



7. Modelo Relacional e Scripts da Controller



7.1 Model Contato

Model Contato - Classe responsável por criar a tabela Contatos no Banco de Dados. Através das DataAnnotations é possível definir o que é Chave Primária [Key], Tipo de Campo [Column(TypeName = "VARCHAR")] , Tamanho Máximo de Caracteres do Campo [MaxLength(60)], e se o mesmo pode ser Nulo [Required(ErrorMessage = "O nome do contato é obrigatório!")].

```
[Table("Contatos")]
13 referências
public class Contato
{
    [Key]
    5 referências
    public int ContatoID { get; set; }

    [MaxLength(60)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Nome do Contato")]
    [Required(ErrorMessage = "O nome do contato é obrigatório!")]
    4 referências
    public string NomeContato { get; set; }

    [MaxLength(14)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Telefone Fixo")]
    0 referências
    public string TelefoneFixoContato { get; set; }

    [MaxLength(14)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Celular")]
    0 referências
    public string CelularContato { get; set; }

    [MaxLength(30)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "E-mail")]
    0 referências
    public string EmailContato { get; set; }

    0 referências
    public virtual ICollection<Endereco> ListaEnderecos { get; set; }

    0 referências
    public virtual ICollection<Tarefa> ListaTarefas { get; set; }
}
```

7.2 Model Endereco

Classe responsável por criar a tabela Enderecos no Banco de Dados.

```
public class Endereco
{
    [Key]
    [Referencia]
    public int EnderecoId { get; set; }

    [Display(Name = "Nome do Contato")]
    [Referencia]
    public int ContatoId { get; set; }

    [MaxLength(9)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Cep")]
    [Required(ErrorMessage = "O Cep é obrigatório!")]
    [Referencia]
    public string CepEndereco { get; set; }

    [MaxLength(40)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Rua")]
    [Required(ErrorMessage = "O nome da rua é obrigatório!")]
    [Referencia]
    public string NomeRua { get; set; }

    [MaxLength(8)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Número")]
    [Referencia]
    public string NumeroEndereco { get; set; }

    [MaxLength(40)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Bairro")]
    [Required(ErrorMessage = "O nome do bairro é obrigatório!")]
    [Referencia]
    public string NomeBairro { get; set; }

    [MaxLength(40)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Cidade")]
    [Required(ErrorMessage = "O nome da cidade é obrigatório!")]
    [Referencia]
    public string NomeCidade { get; set; }

    [MaxLength(2)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "UF")]
    [Required(ErrorMessage = "A unidade federativa é obrigatório!")]
    [Referencia]
    public string UFestado { get; set; }

    [MaxLength(40)]
    [Column(TypeName = "VARCHAR")]
    [Display(Name = "Complemento")]
    [Referencia]
    public string ComplementoEndereco { get; set; }

    [ForeignKey("Contato")]
    [Referencia]
    public virtual Contato objContato { get; set; }
}
```

7.3 Model Tarefa

Responsável por criar a tabela Tarefas no Banco de Dados.

```
[Table("Tarefas")]
15 referências
public class Tarefa
{
    [Key]
    1 referências
    public int TarefaID { get; set; }

    [DisplayName = "Responsável pela Tarefa"]
    0 referências
    public int ContatoID { get; set; }

    [MaxLength(28)]
    [Column(TypeName = "VARCHAR")]
    [DisplayName(Name = "Nome da Tarefa")]
    [Required(ErrorMessage = "O nome da tarefa é obrigatório!")]
    0 referências
    public string NomeTarefa { get; set; }

    [DataType(DataType.Date, ErrorMessage = "Data em formato inválido - DD/MM/AAAA")]
    [DisplayFormat(DataFormatString = "{0:dd/MM/yyyy}", ApplyFormatInEditMode = true)]
    [DisplayName(Name = "Data de Início")]
    [Required(ErrorMessage = "Data de início é obrigatório!")]
    0 referências
    public DateTime DataInicioTarefa { get; set; }

    [DataType(DataType.Date, ErrorMessage = "Data em formato inválido - DD/MM/AAAA")]
    [DisplayFormat(DataFormatString = "{0:dd/MM/yyyy}", ApplyFormatInEditMode = true)]
    [DisplayName(Name = "Data de Término")]
    [Required(ErrorMessage = "Data de término é obrigatório!")]
    0 referências
    public DateTime DataFimTarefa { get; set; }

    [DisplayName(Name = "Descrição da Tarefa")]
    [MaxLength(500)]
    [Column(TypeName = "VARCHAR")]
    0 referências
    public string DescricaoTarefa { get; set; }

    [DisplayName(Name = "Status da Tarefa")]
    [Column(TypeName = "VARCHAR")]
    0 referências
    public string StatusTarefa { get; set; }

    [ForeignKey("ContatoID")]
    0 referências
    public virtual Contato objContato { get; set; }
}
```

7.4 Classe DaoBase

Responsável abrir e fechar a conexão com o Banco de Dados

```
4 referências
public class DaoBase
{
    protected readonly DataContext banco;

    0 referências
    public DaoBase()
    {
        banco = new DataContext();
    }

    0 referências
    public void Dispose()
    {
        banco.Dispose();
    }
}
```

7.5 Classe TarefaDao

Herda de DaoBase, possui os métodos de Busca, Alteração, Exclusão e Gravação. Essa classe recebe as informações da Classe TarefaController. Obs: no projeto para cada Classe de Model existe uma Dao e uma Controller.

```
6 referências
public class TarefaDao : DaoBase
{
    1 referência
    public void Gravar(Tarefa objTarefa)
    {
        banco.Tarefas.Add(objTarefa);
        banco.SaveChanges();
    }
    1 referência
    public void Alterar(Tarefa objTarefa)
    {
        var entry = banco.Entry(objTarefa);
        banco.Set<Tarefa>().Attach(objTarefa);
        entry.State = EntityState.Modified;
        banco.SaveChanges();
    }
    3 referências
    public Tarefa Buscar(int id)
    {
        return banco.Tarefas.FirstOrDefault(t => t.TarefaID == id);
    }
    1 referência
    public void Excluir(int id)
    {
        Tarefa objTarefa = Buscar(id);
        banco.Tarefas.Remove(objTarefa);
        banco.SaveChanges();
    }
    1 referência
    public IEnumerable<Tarefa> Buscar()
    {
        return banco.Tarefas.ToList();
    }
}
```


7.6 Classe TarefaController

Todas as Controllers fazem o intermédio entre as View que possuem as páginas Web e as Models.

```
O referências
public class TarefaController : Controller
{
    // GET: Tarefa
    O referências
    public ActionResult Index()
    {
        var tarefas = new TarefaDao().Buscar();
        return View(tarefas);
    }
    O referências
    public ActionResult Novo()
    {
        ViewBag.Contatos = new SelectList(new ContatoDao().Buscar().ToDictionary(x => x.ContatoID, x => x.NomeContato), "Key", "Value");
        return View();
    }
    O referências
    public ActionResult Editar(int id)
    {
        ViewBag.Contatos = new SelectList(new ContatoDao().Buscar().ToDictionary(x => x.ContatoID, x => x.NomeContato), "Key", "Value");
        var tarefa = new TarefaDao().Buscar(id);
        return View(tarefa);
    }
    [HttpPost]
    O referências
    public ActionResult Editar(Tarefa objTarefa)
    {
        new TarefaDao().Alterar(objTarefa);
        return RedirectToAction("Index", "Tarefa");
    }
    [HttpPost]
    O referências
    public ActionResult Novo(Tarefa objTarefa)
    {
        new TarefaDao().Gravar(objTarefa);
        return RedirectToAction("Index", "Tarefa");
    }
    O referências
    public ActionResult Detalhar(int id)
    {
        Tarefa tarefa = new TarefaDao().Buscar(id);
        return View(tarefa);
    }
    O referências
    public ActionResult Excluir(int id)
    {
        new TarefaDao().Excluir(id);
        return RedirectToAction("Index", "Tarefa");
    }
}
```

8. Links

Vídeo Parte 01:

<https://mega.nz/file/zVFxWaTD#8G8DblxjYm6TV6CLcAohiTwTgU9ymkH0VFsP-yi8nhQ>

Vídeo Parte 02:

<https://mega.nz/file/SAd1WKxY#5Pg5TdpQ2EATfoH2dIYN4hDsFTn2a7YmyEHMygHZwk0>

Projeto Compactado:

<https://mega.nz/file/7dEBEazD#fIDBEPv8iYk-s4TMwZPLkIqBkgezW4e05Ylbvwx1P0>