Tabelas do Banco

public class Usuario

{

[Key]

public int UserId { get; set; }

[Display(Name = "E-Mail")]

[Required(ErrorMessage = "O Campo {0} é Obrigatório!")]

[StringLength(100, ErrorMessage = "O Campo {0} pode ter no máximo {1} e minimo {2} caracteres", MinimumLength = 7)]

[DataType(DataType.EmailAddress)]

[Index("UserNameIndex", IsUnique = true)]

public string UserName { get; set; }

[Display(Name = "Nome")]

[Required(ErrorMessage = " O Campo {0} é Obrigatório!")]

[StringLength(50, ErrorMessage = " O Campo {0} pode ter no máximo {1} e minimo {2} caracteres ", MinimumLength = 2)]

public string Nome { get; set; }

[Display(Name = "Sobrenome")]

[Required(ErrorMessage = " O Campo {0} é Obrigatório!")]

[StringLength(50, ErrorMessage = " O Campo {0} pode ter no máximo {1} e minimo {2} caracteres ", MinimumLength = 2)]

public string Sobrenome { get; set; }

[Display(Name = "Usuário")]

public string NomeCompleto { get { return string.Format("{0} {1}", this.Nome, this.Sobrenome); } }

[Required(ErrorMessage = " O Campo {0} é Obrigatório!")]

[StringLength(20, ErrorMessage = " O Campo {0} pode ter no máximo {1} e minimo {2} caracteres ", MinimumLength = 7)]

public string Telefone{ get; set; }

[Required(ErrorMessage = " O Campo {0} é Obrigatório!")]

[StringLength(100, ErrorMessage = " O Campo {0} pode ter no máximo {1} e minimo {2} caracteres ", MinimumLength = 10)]

public string Endereco { get; set; }

[DataType(DataType.ImageUrl)]

public string Photo { get; set; }

[Display(Name = "É Estudante")]

public bool Estudante { get; set; }

[Display(Name = "É Professor")]

public bool Professor { get; set; }

public virtual ICollection<Group> Grupos { get; set; }

public virtual ICollection<GroupDetail> GruposDetalhes { get; set; }

}

## Grupo

public class Grupo

{

[Key]

public int GrupoId { get; set; }

[Required(ErrorMessage = " O Campo {0} é Obrigatório!")]

[StringLength(50, ErrorMessage = " O Campo {0} pode ter no máximo {1} e minimo {2} caracteres ", MinimumLength = 3)]

[Index("GrupoDescricaoIndex", IsUnique = true)]

public string Descricao { get; set; }

public int UserId { get; set; }

public virtual Usuario Professor { get; set; }

public virtual ICollection<GroupDetail> GroupoDetalhes { get; set; }

}

## Grupo Detalhes

public class GroupoDetalhes

{

[Key]

public int GrupoDetalhesId { get; set; }

public int GrupoId { get; set; }

public int UserId { get; set; }

public virtual Grupo Grupo { get; set; }

public virtual Usuario Estudante { get; set; }

public string GrupoEstudante { get { return string.Format("{0} / {1}", Grupo.Descricao, Estudante.NomeCompleto); } }

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

}

## Notas

public class Notas

{

[Key]

public int NotaId { get; set; }

public int GrupoDetalhesId { get; set; }

[Required(ErrorMessage = "Digite o campo {0}")]

[Range(0, 1, ErrorMessage = "O campo {0} tem que está {1} e {2}")]

[DisplayFormat(DataFormatString = "{0:P2}", ApplyFormatInEditMode = false)]

public float Percentual { get; set; }

[Required(ErrorMessage = "Digite o campo {0}")]

[Range(0, 5, ErrorMessage = "O campo {0} contém valores entre {1} and {2}")]

[DisplayFormat(DataFormatString = "{0:N2}", ApplyFormatInEditMode = false)]

public float Nota { get; set; }

public virtual GrupoDetalhes GrupoDetalhes { get; set; }

}