

UNIJOBS

Integrantes

DUCILIO MANJATE

MILENE DANELLI

NELSON BAEZ

ENTREGA 1

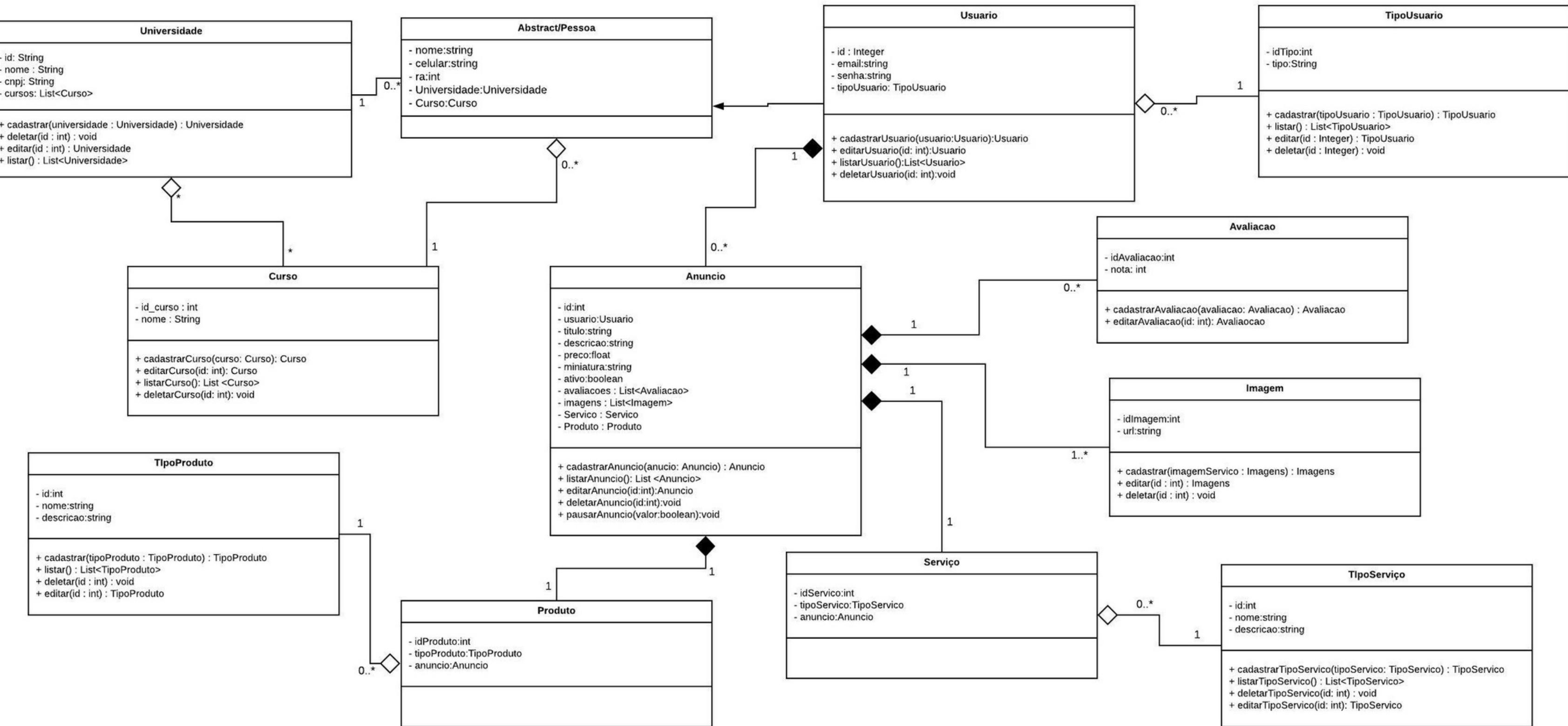
DIAGRAMAS - UML e ER
SQL & JAVA

01



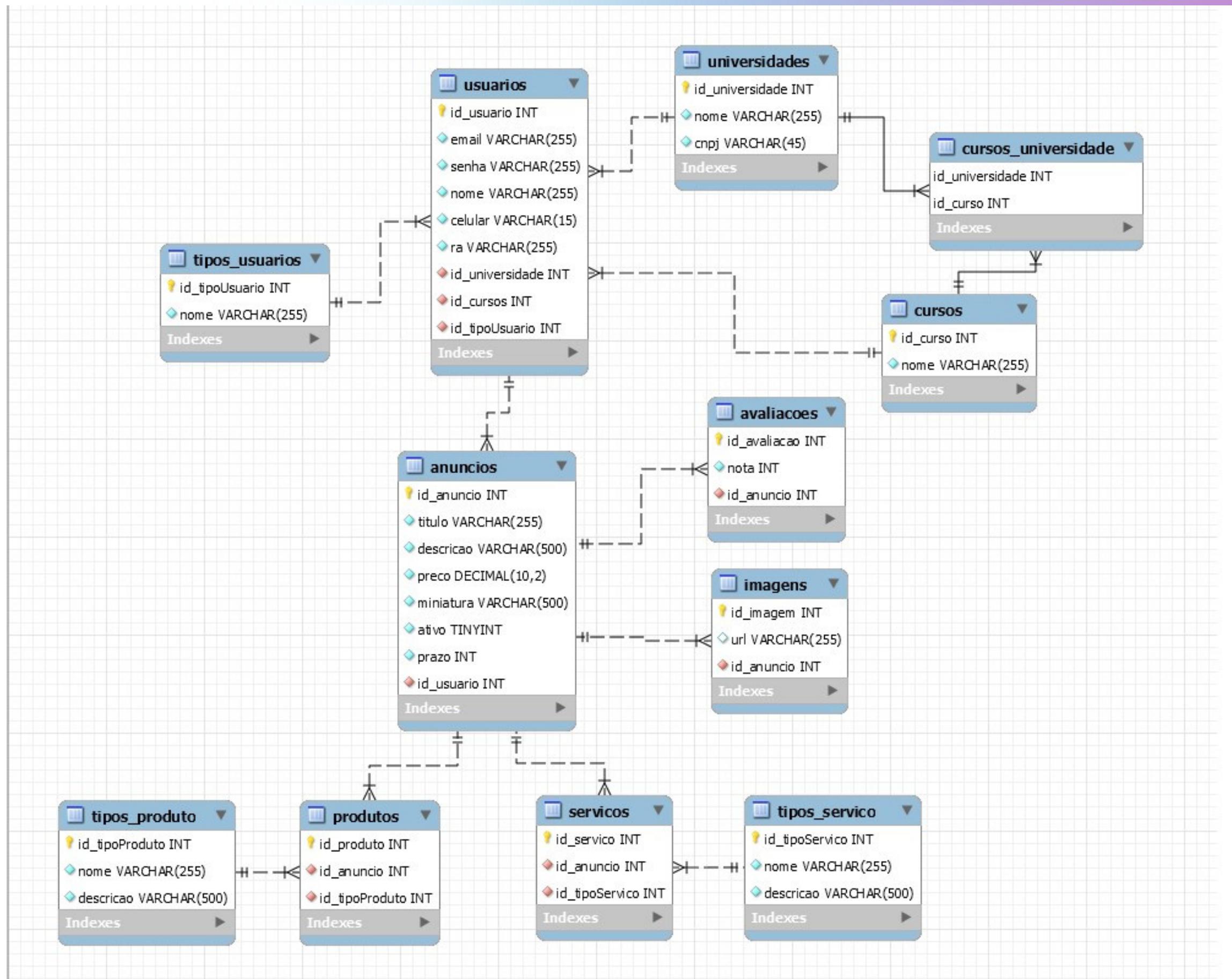
Diagramas Classe

02



Diagramas entidade relacionamento

02



```
22  
23 • CREATE TABLE IF NOT EXISTS unijobs.usuarios (  
24     id_usuario INT NOT NULL AUTO_INCREMENT,  
25     email VARCHAR(255) NOT NULL,  
26     senha VARCHAR(255) NOT NULL,  
27     nome VARCHAR(255) NOT NULL,  
28     celular VARCHAR(15) NOT NULL,  
29     ra VARCHAR(255) NOT NULL,  
30     id_universidade INT NOT NULL,  
31     id_cursos INT NOT NULL,  
32     id_tipoUsuario INT NOT NULL,  
33     PRIMARY KEY (id_usuario),  
34     FOREIGN KEY (id_universidade)  
         REFERENCES unijobs.universidades (id_universidade)  
         ON DELETE RESTRICT ON UPDATE CASCADE,  
35     FOREIGN KEY (id_cursos)  
         REFERENCES unijobs.cursos (id_curso)  
         ON DELETE RESTRICT ON UPDATE CASCADE,  
36     FOREIGN KEY (id_tipousuario)  
         REFERENCES unijobs.tipos_usuarios (id_tipoUsuario)  
         ON DELETE RESTRICT ON UPDATE CASCADE  
37 );
```

```
CREATE TABLE IF NOT EXISTS unijobs.tipos_produto (  
    id_tipoProduto INT NOT NULL AUTO_INCREMENT,  
    nome VARCHAR(255) NOT NULL,  
    descricao VARCHAR(500) NOT NULL,  
    PRIMARY KEY (id_tipoProduto)  
);  
  
CREATE TABLE IF NOT EXISTS unijobs.cursos_universidade (  
    id_universidade INT NOT NULL,  
    id_curso INT NOT NULL,  
    PRIMARY KEY (id_universidade , id_curso),  
    FOREIGN KEY (id_universidade)  
        REFERENCES unijobs.universidades (id_universidade)  
        ON DELETE RESTRICT ON UPDATE CASCADE,  
    FOREIGN KEY (id_curso)  
        REFERENCES unijobs.cursos (id_curso)  
        ON DELETE RESTRICT ON UPDATE CASCADE  
);  
  
CREATE TABLE IF NOT EXISTS unijobs.imagens (  
    id_imagem INT NOT NULL AUTO_INCREMENT,  
    url VARCHAR(255) NULL,  
    id_anuncio INT NOT NULL,  
    PRIMARY KEY (id_imagem),  
    FOREIGN KEY (id_anuncio)
```



```
use Unijobs;

insert into cursos (nome) values('Engenharia de Software'), ('Arquitetura'), ('Direito'), ('Contabilidade'), ('Administrador');
select * from cursos;

insert into universidades (nome, cnpj) values ('Uniamerica', '18715633000141'), ('UTFPR', '75101873000190');
select * from universidades;

insert into cursos_universidade(id_universidade, id_curso) values (1, 1), (1, 2), (1, 3), (2, 3), (2, 4), (2, 5);
select * from cursos_universidade;

select * from universidades
join cursos_universidade using (id_universidade)
join cursos using (id_curso)
order By universidades.nome;

insert into tipos_usuarios(nome) values ('Administrador Sistema'), ('Administrador Universidade'), ('Vendedor');
select * from tipos_usuarios;

insert into usuarios(email, senha, nome, celular, ra, id_universidade, id_cursos, id_tipousuario)
VALUES
('nelson@gmail.com.br', 'tomaladaca', 'Nelson Baez', '988322026', 666666, 1, 1, 1),
('milene@gmail.com.br', 'testeteste', 'Milene Danelli', '564654561', 541364, 2, 1, 1),
('lio@gmail.com.br', 'jotajota', 'Lio Croons', '564877123', 123457, 1, 2, 2),
('jorgin@gmail.com.br', 'jorginhoemprestaa12', 'Jorge', '123123123', 444466, 2, 2, 2),
('seuze@gmail.com.br', 'nomedafilha123', 'Seu Zé', '77777777', 777777, 1, 1, 3);

Ativar o Windo
select usuarios.*. universidades.nome as universidade. cursos.nome as curso. tipos_usuarios.nome as tipo_usuario
```



```
use Unijobs;

insert into cursos (nome) values('Engenharia de Software'), ('Arquitetura'), ('Direito'), ('Contabilidade'), ('Administrador');
select * from cursos;

insert into universidades (nome, cnpj) values ('Uniamerica', '18715633000141'), ('UTFPR', '75101873000190');
select * from universidades;

insert into cursos_universidade(id_universidade, id_curso) values (1, 1), (1, 2), (1, 3), (2, 3), (2, 4), (2, 5);
select * from cursos_universidade;

select * from universidades
join cursos_universidade using (id_universidade)
join cursos using (id_curso)
order by universidades.nome;

insert into tipos_usuarios(nome) values ('Administrador Sistema'), ('Administrador Universidade'), ('Vendedor');
select * from tipos_usuarios;

insert into usuarios(email, senha, nome, celular, ra, id_universidade, id_cursos, id_tipousuario)
VALUES
('nelson@gmail.com.br', 'tomaladaca', 'Nelson Baez', '988322026', 666666, 1, 1, 1),
('milene@gmail.com.br', 'testetesteste', 'Milene Danelli', '564654561', 541364, 2, 1, 1),
('lio@gmail.com.br', 'jotajota', 'Lio Croons', '564877123', 123457, 1, 2, 2),
('jorgin@gmail.com.br', 'jorginhoemprestaa12', 'Jorge', '123123123', 444466, 2, 2, 2),
('seuze@gmail.com.br', 'nomedafilha123', 'Seu Zé', '77777777', 777777, 1, 1, 3);

Ativar o Windo
select usuarios.*. universidades.nome as universidade. cursos.nome as curso. tipos_usuarios.nome as tipo_usuario
```



SQL

Query 1 Unijobs Inserts e Selects Prod e... Unijobs Create Database prods...

26 ('seuze@gmail.com.br', 'nomedafilha123', 'Seu Zé', '77777777', 777777, 1, 1, 3);
27
28 • select usuarios.*, universidades.nome as universidade, cursos.nome as curso, tipos_usuarios.nome as tipo_usuario
29 from usuarios
30 join universidades using (id_universidade)
31 join cursos on id_curso = id_cursos
32 join tipos_usuarios using(id_tipousuario);
33
34 • select * from tipos_produto;
35 • insert into tipos_produto (nome, descricao)
36 values
37 ('Eletrônicos', 'celulares, computadores, câmeras, caixas de som'),
38 ('Roupas', 'Camisas, sapatos, calças');
39
40 • select * from tipos_servico;
41 • insert into tipos_servico (nome, descricao)
42 values
43 ('Manutenção em Computadores', 'celulares, computadores, câmeras, caixas de som');
44
45 select * from anuncios;
46 • INSERT INTO anuncios (titulo, descricao, preco, miniatura, ativo, prazo, id_usuario)

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid Form Editor

	id_usuario	email	senha	nome	celular	ra	id_universidade	id_cursos	id_tipoUsuario	universidade	curso	tipo_usuario
▶	1	nelson@gmail.com.br	tomaladaca	Nelson Baez	988322026	666666	1	1	1	Uniamerica	Engenharia de Software	Administrador Sistema
	3	lio@gmail.com.br	jotajota	Lio Croons	564877123	123457	1	2	2	Uniamerica	Arquitetura	Administrador Universidade
	5	seuze@gmail.com.br	nomedafilha123	Seu Zé	77777777	777777	1	1	3	Uniamerica	Engenharia de Software	Vendedor
	2	milene@gmail.com.br	testetestete	Milene Danelli	564654561	541364	2	1	1	UTFPR	Engenharia de Software	Administrador Sistema
	4	jorgin@gmail.com.br	jorginhoemprestaa12	Jorge	123123123	444466	2	2	2	UTFPR	Arquitetura	Administrador Universidade

Result 27 x Ativar o Window Acesse Configurações Read Only

03



SQL

03

```
51 ('Formatação de PC', 'Formato computadores, notebooks etc...', 50.00, 'umlinkqualquer', 1, 7, 1);
52
53 • insert into produtos(id_anuncio, id_tipoProduto)
54 VALUES
55 (1, 2),
56 (2, 2),
57 (3, 1);
58
59 • insert into servicos(id_anuncio, id_tipoServico)
60 VALUES
61 (4, 1);
62
63
64 • select * from produtos
65 left join anuncios using (id_anuncio)
66 join usuarios using(id_usuario)
67 where id_universidade = 1
68 ;
69 select * from servicos left join anuncios using (id_anuncio);
70
71
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	id_usuario	id_anuncio	id_produto	id_tipoProduto	titulo	descricao	preco	miniatura	ativo	prazo	email	senha	nome	celular	ra	id_universidade	id_c
▶	3	2	2	2	Bermuda	Bermudas jeans, moletom.	60.00	https://www.google.com/ur	1	15	lio@gmail.com.br	jotajota	Lio Croons	564877123	123457	1	2
	1	3	3	1	Jogos	de ps4, xbox usados	120.00	https://www.google.com/imgres?imgurl=https...	1	5	nelson@gmail.com.br	tomaladaca	Nelson Baez	988322026	666666	1	1

Result Grid | Form Editor | Ativar o Window | Acesse Configurações | Read Only

Result 28 ×

The screenshot shows a Java IDE interface with a dark theme. The left sidebar displays a project structure with packages: connection, controller, model, and web. The controller package contains several controller classes like AnuncioController, AvacacaoController, CursoController, etc. The model package contains entity classes like Anuncio, Avaliacao, Curso, Imagem, Pessoa, Produto, Servico, etc. The web package contains HttpHandler classes for various endpoints.

The main editor window shows the `Application.java` file with the following code:

```
public class Application {
    public static void main(String[] args) {
        // TODO Auto-generated method stub

        Curso curso1 = new Curso(id: 1, nome: "Engenharia Software");
        Curso curso2 = new Curso(id: 2, nome: "Direito");
        Curso curso3 = new Curso(id: 3, nome: "Medicina");
        ArrayList<Curso> cursos = new ArrayList();

        cursos.add(curso1);
        cursos.add(curso2);
        cursos.add(curso3);

        Universidade universidade1 = new Universidade(id: 1, nome: "Uniamerica", cnpj: "88888888888888", cursos);

        TipoUsuario tipoUsuario1 = new TipoUsuario(id: 1, tipo: "Administrador Geral");

        Usuario usuario1 = new Usuario(nome: "Nelson", celular: "888888888", ra: 888888, universidade1, curso1, id: 1,
            email: "teste@gmail.com", senha: "sensenha", tipoUsuario1);

        Usuario usuario2 = new Usuario(nome: "Milene", celular: "77777", ra: 7777, universidade1, curso1, id: 7,
            email: "mi@gmail.com", senha: "123", tipoUsuario1);

        TipoProduto tipoProduto = new TipoProduto(id: 1, nome: "Gastronomia", descricao: "Produtos relacionados a comestiveis");

        ArrayList<Imagen> imagens = new ArrayList();
        ArrayList<Avaliacao> avaliacoes = new ArrayList();

        Produto produto = new Produto(id: 1, usuario2, titulo: "Rocambole", descricao: "Sabor coco com chocolate", preco: 20.99,
            miniatura: "https://www.google.com/url?sa=i&url=https%3", ativo: true, avaliacoes, imagens, id1: 1, tipoProduto);

        System.out.println(produto);
    }
}
```

The run tab at the bottom shows the output of the application:

```
F:\Java\jdk1.8.0_281\bin\java.exe ...
Produto{id=1, tipoProduto=TipoProduto{id=1, nome='Gastronomia', descricao='Produtos relacionados a comestiveis'}, titulo=Rocambole, descricao=Sabor coco com chocolate, preco=20.99, miniatura=https://www.google.com/url?sa=i&url=h...
Process finished with exit code 0
```

```
public class Produto extends Anuncio{
    private Integer id;
    private TipoProduto tipoProduto;

    public Produto() {
    }

    public Produto(Integer id, Usuario usuario, String titulo, String descricao, Double preco, String miniatura,
                   Boolean ativo, ArrayList<Avaliacao> avaliacoes, ArrayList<Imagem> imagens, Integer id1,
                   TipoProduto tipoProduto) {
        super(id, usuario, titulo, descricao, preco, miniatura, ativo, avaliacoes, imagens);
        this.id = id1;
        this.tipoProduto = tipoProduto;
    }

    public Produto(Integer id) {
        this.id = id;
    }

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    @Override
    public String toString() {
        return "Produto{" +
            "id=" + id +
            ", tipoProduto=" + tipoProduto +
            ", titulo=" + getTitulo() +
            ", descricao=" + getDescricao() +
            ", preco=" + getPreco() +
            ", miniatura=" + getMiniatura() +
            ", ativo=" + getAtivo() +
            ", avaliacoes=" + getAvaliacoes() +
            ", imagens=" + getImagens() +
            ", id1=" + getId() +
            '}';
    }
}
```



```
public class Anuncio {  
    private Integer id;  
    private Usuario usuario;  
    private String titulo;  
    private String descricao;  
    private Double preco;  
    private String miniatura;  
    private Boolean ativo;  
    private ArrayList<Avaliacao> avaliacoes;  
    private ArrayList<Imagem> imagens;  
  
    public Anuncio() {}  
  
    public Anuncio(Integer id, Usuario usuario, String titulo, String descricao, Double preco, String miniatura,  
        Boolean ativo, ArrayList<Avaliacao> avaliacoes, ArrayList<Imagem> imagens) {  
        this.id = id;  
        this.usuario = usuario;  
        this.titulo = titulo;  
        this.descricao = descricao;  
        this.preco = preco;  
        this.miniatura = miniatura;  
        this.ativo = ativo;  
        this.avaliacoes = avaliacoes;  
        this.imagens = imagens;  
    }  
  
    public Integer getId() { return id; }  
  
    public void setId(Integer id) { this.id = id; }  
  
    public Usuario getUsuario() { return usuario; }  
  
    public void setUsuario(Usuario usuario) { this.usuario = usuario; }
```



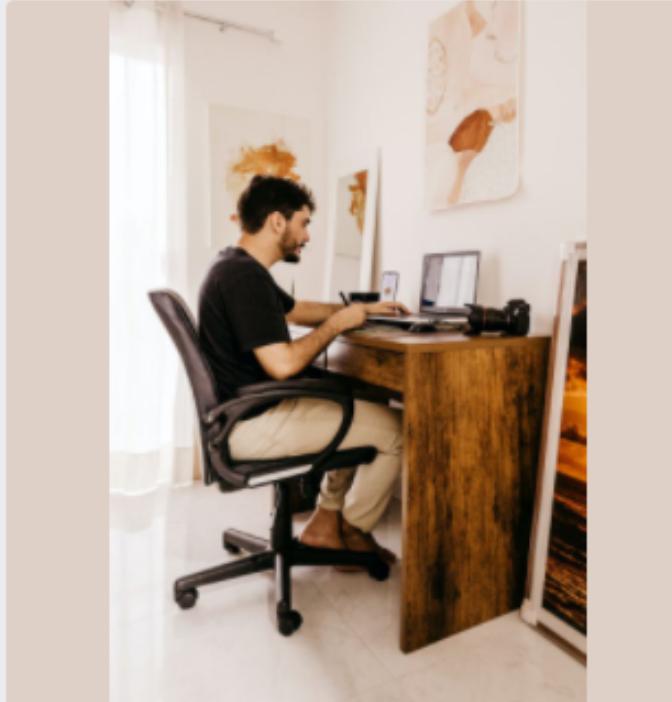
Quadros Ir para...

Quadro Projeto acadêmico -PI star space apps Free Visível para o Time M MD +1 Convidar

Trello

Estudos


Perceber o funcionamento da linguagem Orientada a Objeto


Instalação e configuração do

Reuniões todas as segundas

29/03/20: Segunda-Feira

<https://github.com/savage-squad/Unijobs-BackEnd.git>

+ Adicionar outro cartão

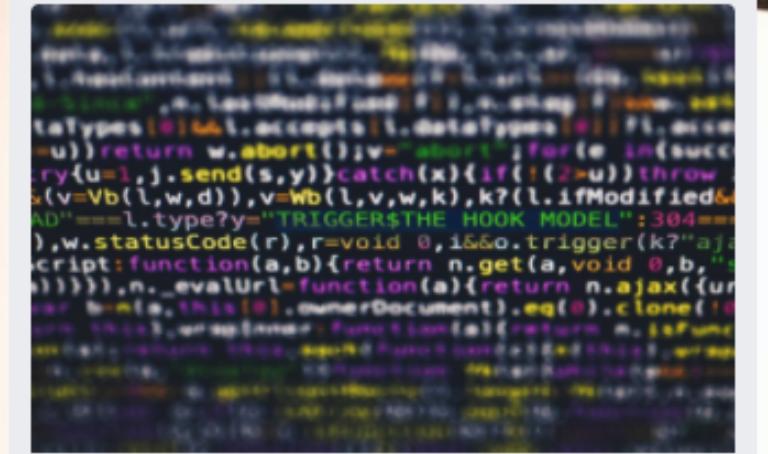
Por Fazer


Estudo do Java Puro, Mysql, documentação: Postman/Insomnia para a construção da API ou MVC e JWT

Fazendo

Documentação

1 comment, 2 attachments


Model

1 attachment

Finalizado

08/03/20: Segunda-Feira

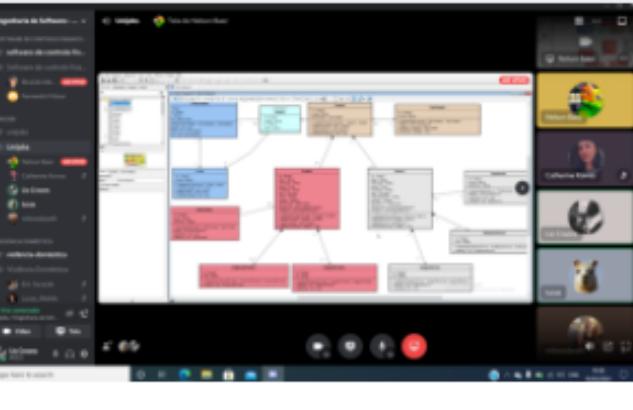
1 attachment

Diagrama ER

8 de mar 4 attachments

Diagrama de classe

2 de mar 3 comments, 6 attachments


15/03/20: Segunda-Feira

1 attachment

Diagrama de caso de uso


Controller

+ Adicionar outro cartão

Ajustar as telas do front-end

Diagrama de Seqüência

+ Adicionar outro cartão

Butler (4 Tips) Google Drive



Type here to search





savage-squad

Projectos académicos da UniAmérica

📍 Uni 🐦 @Uni

Repositories 3

Packages

People 4

Teams

Projects

Settings

Pinned repositories

Customize pinned repositories

Unijobs-BackEnd

Project integrador UniAmerica

Java

unijobs-front

Forked from viniengelage/unijobs-front

JavaScript

UnijobsPuro-BackEnd

Java

Find a repository...

Type ▾

Language ▾

Sort ▾

New

UnijobsPuro-BackEnd

Java 0 ⚡ 0 ⭐ 0 ⚠ 0 Updated 1 hour ago

Top languages

Java JavaScript

Unijobs-BackEnd

Project integrador UniAmerica

Java 0 ⚡ 0 ⭐ 0 ⚠ 0 Updated 6 days ago

People 4 >

Invite someone

unijobs-front

Forked from viniengelage/unijobs-front

JavaScript 5 ⚡ 0 ⭐ 0 ⚠ 0 Updated 12 days ago