

INSTITUO PROA
PROFESSÃO – TÉCNICO

CAMILY VITÓRIA, DIEGO SANTOS, GIULLIA SILVA, HENRIQUE ARAUJO,
IGOR VIANA, LUCAS WALDES, MATHEUS SILVERIO, PEDRO OLIVEIRA

DOCUMENTAÇÃO DO BANCO DE DADOS
BANCO DE DADOS DO “MENINA, NEM TE CONTO!”

SÃO PAULO – SP

2022

Sumário

1.	USO DE DADOS	3
2.	DICIONARIO DE CLASSES.....	4
2.1.	Identificação Geral	4
2.2.	Fotos identificação geral.....	5
2.3.	Vagas	5
2.4.	Fotos vagas	6
2.5.	Propostas.....	6
2.6.	Newsletter	6
2.7.	Cursos	7
2.8.	Capa cursos.....	7
2.9.	Categoria cursos.....	7
2.10.	Histórico Login	8
2.11.	Cadastro Freelas	8
2.12.	Certificados.....	9
3.	DIAGRAMA LÓGICO	10
4.	Código em SQL.....	12
4.1.	Uso Interno	12
4.2.	Uso Externo	13
5.	OBSERVAÇÕES GERAIS.....	16

1. USO DE DADOS

De acordo com a Lei Geral de Proteção de Dados nº14.010/2020 que passaram a ser elegíveis em 1º de agosto de 2021, afirmamos que todos os dados pessoais presentes neste Banco de Dados serão utilizados com o único intuito de identificação de usuários, para que possamos garantir segurança a todos.

Os dados de solicitados são:

- Nome
- Sobrenome
- Nome Social (Caso possua)
- CPF
- Data de Nascimento
- MEI (pessoas mulheres cadastradas como freelancer)

2. DICIONARIO DE CLASSES

2.1. Identificação Geral

identificacao_geral				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary Key
nome	Armazena o nome do Usuario	CHARACTER	70	NOT NULL
sobrenome	Armazena o sobrenome do Usuario	CHARACTER	70	NOT NULL
nome_social	Armazena o nome social do usuario caso exista	CHARACTER	70	
cpf	Armazena o CPF do Usuario e deve ser único no banco de dados	CHARACTER	11	UNIQUE
dt_nascimento	Armazena a Data de nascimento do usuario	DATE	10	NOT NULL
telefone	Armazena o Telefone do Usuario	CHARACTER	20	NOT NULL
cep	Armazena o CEP do Usuario	CHARACTER	9	NOT NULL
email	Armazena Email do usuario que vai ser utilizado para fazer login e ele deve ser único	CHARACTER	255	UNIQUE , NOT NULL
senha	Armazena Senha do Usuario que vai ser utilizada para fazer login e ela vai para o banco já criptografada	CHARACTER	255	NOT NULL
dt_cadastro	Recebe a Data de cadastro do usuario pegando diretamente do sistema	DATE	10	NOT NULL
status	Recebe o Status se a conta está ativa ou não	BOOLEAN	10	DEFAULT = TRUE
nivel_acesso	Recebe o nível de acesso do Usuario, os níveis podem ser CONTR para contratante, FREE para freelancers e ADMIN para os administradores do site	VARCHAR	5	NOT NULL

2.2. Fotos identificação geral

fotos_iden_ger				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANH	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary key
caminho_ft	Armazena a Imagem do Usuario no banco	NUMERICO	1000	NOT NULL
fk_iden	Recebe o valor que a imagem está associada na tabela Identificação Geral	NUMERICO	50	FK - Foreign key

2.3. Vagas

vagas				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary key
titulo_vaga	Armazena o Titulo da publicação	CARACTER	255	NOT NULL
dt_postagem	Armazena o Dia da postagem da publicação	DATE		NOT NULL
hr_postagem	Armazena a Hora da posta da publicação	DATE		NOT NULL
descricao	Armazena a descrição da postagem	CARACTER	10000	NOT NULL
pagamento	Armazena o valor do pagamento	INTEIRO		NOT NULL
urgencia	Armazena a Urgencia para executar o	CARACTER	255	NOT NULL
status	Armazena o status da vaga sendo 0 = fechada ou 1 =	NUMERICO	1	NOT NULL
fk_iden_ger	Mostra quem foi a pessoa que faz a publicação da	NUMERICO	50	NOT NULL

2.4. Fotos vagas

fotos_vaga				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANH	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary key
caminho_ft	Armazena a foto relacionada a vaga	NUMERICO	10000	NOT NULL
fk_vaga	Mostra a qual vaga a foto pertence	NUMERICO	50	NOT NULL

2.5. Propostas

propostas				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary key
propostas	Armazena a proposta feita	INTEIRO		NOT NULL
fk_freelas	Mostra que freelancer fez a proposta	NUMERICO	50	NOT NULL
fk_vagas	Mostra a qual vaga aquela proposta está	NUMERICO	50	NOT NULL

2.6. Newsletter

newsletter				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK- Primary Key
email	Armazena o email que é cadastrado no newsletter	CARACTER	255	NOT NULL
dt_cadastro	Armazena a Data em que pessoa foi cadastrada no newsletter	DATE		NOT NULL

2.7. Cursos

cursos				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANH	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK- Primary Key
nome_curso	Armazena o nome do curso	CARACTER	255	NOT NULL
instituicao	Armazena a instituição que o curso está presente	CARACTER	255	NOT NULL
valor	Armazena o valor do curso	NUMERICO		NOT NULL
tipo_formacao	Armazena o tipo de formacao do curso	CARACTER		NOT NULL
fk_categorias	Mostra a qual categoria o curso pertence	NUMERICO	50	Fk - Foreign Key

2.8. Capa cursos

capa_cursos				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANH	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK- Primary Key
caminho_ft	Armazena o caminho da foto	NUMERICO		
fkey_cursos	Mostra a qual curso a imagem pertence	NUMERICO	50	Fk - Foreign Key

2.9. Categoria cursos

categorias_cursos				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK- Primary Key
nome_cate	Armaneza o nome da Categoria de	CARACTER	255	

2.10. Histórico Login

hist_logins				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary key
dt_login	Armazena quando o usuario fez o login na plataforma	DATE		NOT NULL
fk_iden_login	Recebe o valor que o login está associado na tabela Identificação Geral	NUMERICO	50	FK - Foreign key

2.11. Cadastro Freelas

cadastro_freelas				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
ID	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary Key
mei	Armazena o MEI do Usuario Freela e é único	CARACTER	11	UNIQUE
area_formacao	Armazena a Área de formação do Usuario Freela	CARACTER	255	
curriculo	Armazena o Currículo da Usuario Freela	NUMERICO	10000	
fk_idenGer	Recebe o valor que mostra a qual tupla de identificacao geral o perfil está ligado	NUMERICO	50	Fk - Foreign Key
sobre-mim	Armezena o sobre da freelancer	CARACTER	1000	

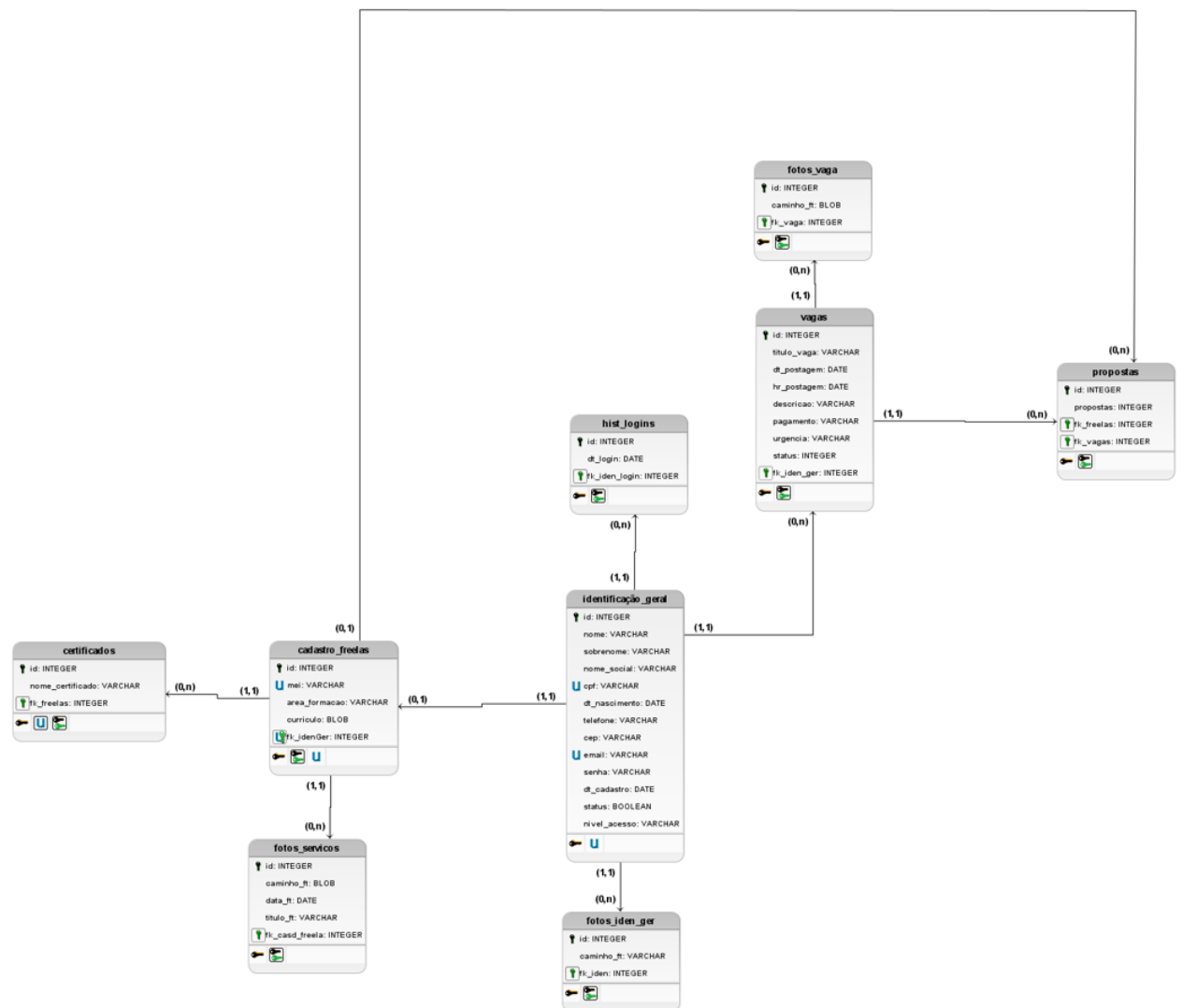
2.12. Certificados

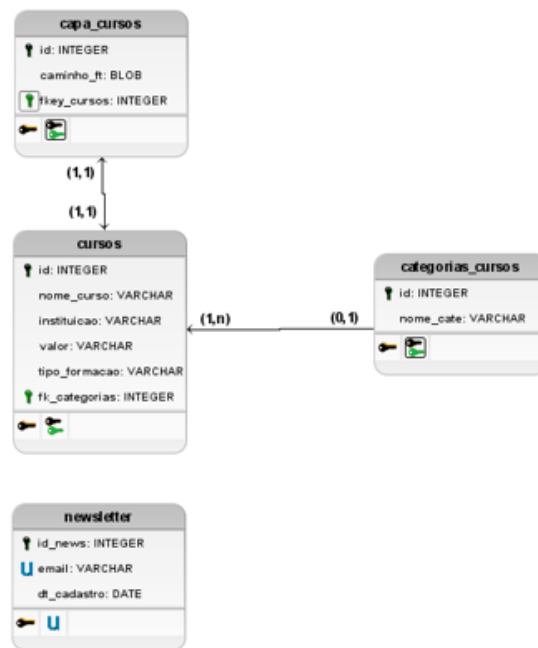
certificados				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
id	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary key
nome_certificado	Armazena o nome do certificado	VARCHAR	255	NOT NULL
fk_freelas	Mostra a qual tupla na tabela cadastro_freelas os certificados pertencem	NUMERICO	50	Fk - Foreign Key

2.13. Fotos Serviços

fotos_servicos				
ATRIBUTOS	DESCRIÇÃO	TIPO	TAMANHO	RESTRIÇÃO
ID	Armazena o ID que é gerado automaticamente	NUMERICO	50	PK - Primary Key
caminho_ft	Armazena foto de serviços do Usuario	NUMERICO	1000	NOT NULL
data_ft	Armazena a data que o serviço foi feito	DATE		
titulo_ft	Armazena o titulo que vai estar presente na apresentação do serviço	CARACTER	1000	NOT NULL
fk_casd_freela	Recebe o valor que mostra a qual tupla de identificação geral as fotos pertencem	NUMERICO	50	NOT NULL

3. DIAGRAMA LÓGICO





4. Código em SQL

4.1. Uso Interno

```
CREATE TABLE IF NOT EXISTS newsletter(  
    id_news          INTEGER PRIMARY KEY  
    AUTO_INCREMENT,  
    email            VARCHAR(500) UNIQUE,  
    dt_cadastro      DATE  
);
```

```
CREATE TABLE IF NOT EXISTS categorias_cursos(  
    id                INTEGER PRIMARY KEY,  
    nome_cate         VARCHAR(255) NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS cursos(  
    id                INTEGER PRIMARY KEY,  
    nome_curso        VARCHAR(255) NOT NULL,  
    instituicao        VARCHAR(255) NOT NULL,  
    valor              VARCHAR(255) DEFAULT 'GRATUITO',  
    tipo_formacao      VARCHAR(255) NOT NULL,  
    fk_categorias     INTEGER  
);
```

```
CREATE TABLE IF NOT EXISTS capa_cursos(  
    id                INTEGER PRIMARY KEY,  
    caminho_ft        LONGBLOB,  
    fkey_cursos       INTEGER  
);
```

```
ALTER TABLE CURSOS ADD CONSTRAINT fkey_categorias FOREIGN  
KEY(fk_categorias) REFERENCES categorias_cursos(id);  
ALTER TABLE capa_cursos ADD CONSTRAINT capa_cursos FOREIGN  
KEY(fkey_cursos) REFERENCES cursos(id);
```

4.2. Uso Externo

```
CREATE TABLE IF NOT EXISTS identificacao_geral (  
id          INTEGER PRIMARY KEY AUTO_INCREMENT,  
nome        VARCHAR(255) NOT NULL,  
sobrenome   VARCHAR(255) NOT NULL,  
nome_social VARCHAR(255),  
cpf         VARCHAR(11) NOT NULL UNIQUE,  
dt_nascimento DATE NOT NULL,  
telefone    VARCHAR(50) NOT NULL,  
cep         VARCHAR(50) NOT NULL,  
email       VARCHAR(255) UNIQUE NOT NULL,  
senha       VARCHAR(255) NOT NULL,  
dt_cadastro DATE NOT NULL,  
status      BIT  
);
```

```
CREATE TABLE IF NOT EXISTS hist_logins (  
id          INTEGER PRIMARY KEY AUTO_INCREMENT,  
dt_login    DATE NOT NULL,  
fk_iden_login INTEGER NOT NULL  
);
```

```
ALTER TABLE hist_logins ADD CONSTRAINT fkey_login_hist FOREIGN  
KEY (fk_iden_login) REFERENCES identificacao_geral(id);
```

```
CREATE TABLE IF NOT EXISTS fotos_iden_ger(  
id          INTEGER PRIMARY KEY AUTO_INCREMENT,  
caminho_ft  LONGBLOB NOT NULL,  
fk_iden     INTEGER NOT NULL  
);
```

```
ALTER TABLE fotos_iden_ger ADD CONSTRAINT fkey_fotos_iden  
FOREIGN KEY (fk_iden) REFERENCES identificacao_geral(id);
```

```
CREATE TABLE IF NOT EXISTS cadastro_freelas(  
id          INTEGER PRIMARY KEY AUTO_INCREMENT,  
mei         VARCHAR(50),  
area_formacao VARCHAR(255),  
sobremim VARCHAR(255),  
fk_iden_ger INTEGER NOT NULL UNIQUE  
);
```

```
ALTER TABLE cadastro_freelas ADD CONSTRAINT fkey_cadastro_iden  
FOREIGN KEY (fk_iden_ger) REFERENCES identificacao_geral(id);
```

```
CREATE TABLE fotos_servicos (
```

```

id                INTEGER PRIMARY KEY AUTO_INCREMENT,
caminho_ft        LONGBLOB NOT NULL,
data_ft          DATE NOT NULL,
titulo_ft         VARCHAR(1000),
fk_casd_freela    INTEGER NOT NULL
);

```

```

ALTER TABLE fotos_servicos ADD CONSTRAINT fkey_fotos_servicos
FOREIGN KEY (fk_casd_freela) REFERENCES cadastro_freelas(id);

```

```

CREATE TABLE IF NOT EXISTS certificados(
id                INTEGER PRIMARY KEY AUTO_INCREMENT,
nome_certificado VARCHAR(500) NOT NULL,
fk_freelas       INTEGER NOT NULL
);

```

```

ALTER TABLE certificados ADD CONSTRAINT fkey_certificados FOREIGN
KEY (fk_freelas) REFERENCES cadastro_freelas(id);

```

```

CREATE TABLE vagas (
id                INTEGER PRIMARY KEY
AUTO_INCREMENT,
titulo_vaga       VARCHAR(2000) NOT NULL,
dt_postagem       DATE NOT NULL,
hr_postagem       DATE NOT NULL,
descricao         VARCHAR(10000) NOT NULL,
pagamento        VARCHAR(255) NOT NULL,
urgencia          VARCHAR(255) NOT NULL,
status           BINARY DEFAULT 1,
fk_iden_ger       INTEGER NOT NULL
);

```

```

ALTER TABLE vagas ADD CONSTRAINT fkey_vagas FOREIGN KEY
(fk_iden_ger) REFERENCES identificacao_geral(id);

```

```

CREATE TABLE fotos_vaga (
id                INTEGER PRIMARY KEY AUTO_INCREMENT,
caminho_ft        LONGBLOB NOT NULL,
fk_vaga           INTEGER NOT NULL
);

```

```

ALTER TABLE fotos_vaga ADD CONSTRAINT fkey_fotos_vagas FOREIGN
KEY (fk_vaga) REFERENCES vagas(id);

```

```

CREATE TABLE propostas(
id                INTEGER PRIMARY KEY AUTO_INCREMENT,
proposta         NUMERIC(65,2) NOT NULL,

```

```
fk_freelas          INTEGER NOT NULL,  
fk_vagas             INTEGER NOT NULL  
);
```

```
ALTER TABLE propostas ADD CONSTRAINT fkey_freelas_propostas  
FOREIGN KEY (fk_freelas) REFERENCES cadastro_freelas(id);  
ALTER TABLE propostas ADD CONSTRAINT fkey_vagas_propostas  
FOREIGN KEY (fk_vagas) REFERENCES vagas(id);
```

5. OBSERVAÇÕES GERAIS

Todos os arquivos, diagramas e tabelas estarão anexadas a pasta desta documentação.