

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ

Banco de Dados – PJBL Fase 2

KindAct

Membros:

João Victor - 40102065

Guilherme Reis Carvalho – 40093122

Nicolas Lobo - 40090033

Vinicius Trevisan - 40100664

Curitiba

2024

Sumário

Descrição do Projeto	2
Desafios	2
Diagrama de Entidade e Relacionamento.....	3
Lista de Requisitos e Comandos SQL	5
VIEWS, PROCEDURES E FUNCTIONS	11

*** NOTA: alteramos alguns requisitos do projeto. O código também foi alterado, com a criação de views, e mais tabelas necessárias para garantir o funcionamento dos requisitos.

** COPIE E COLE NO GOOGLE PARA VER O CODIGO INTEIRO:

<https://github.com/ViniMTrevisan/PUC-PR---Segundo-semester/blob/main/banco%20de%20dados/programacao/kindact.sql>

Descrição do Projeto

O nosso projeto visa abrir oportunidades para ONGs e voluntários ao redor do Brasil, para que pessoas necessitadas sejam cada vez mais acessíveis e ajudadas para quem quer fazer a diferença. Nosso projeto conectará ONGs com Voluntários em um só site, onde ambos poderão se conectar com o outro, o que reduzirá a dificuldade das ONGs acharem ajuda e dos Voluntários acharem ONGs que se encaixem com o perfil deles.

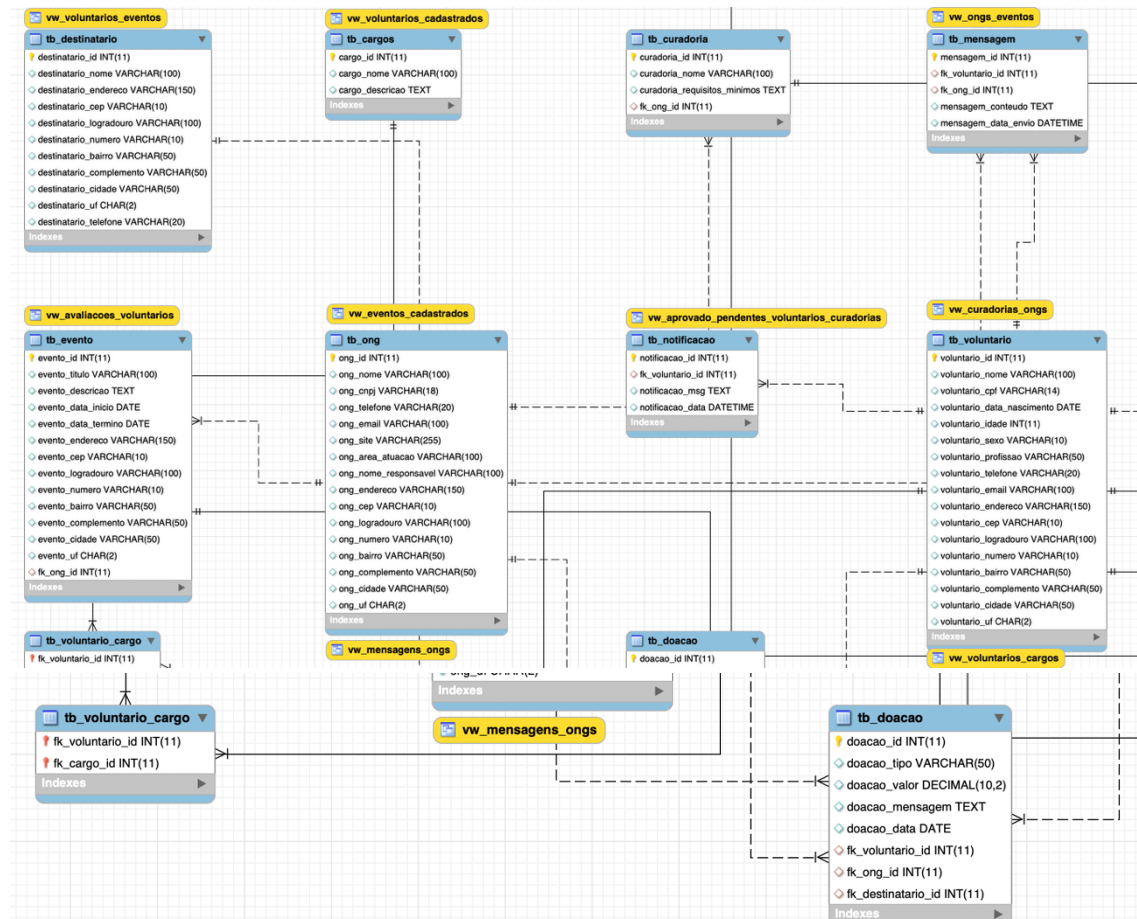
Desafios

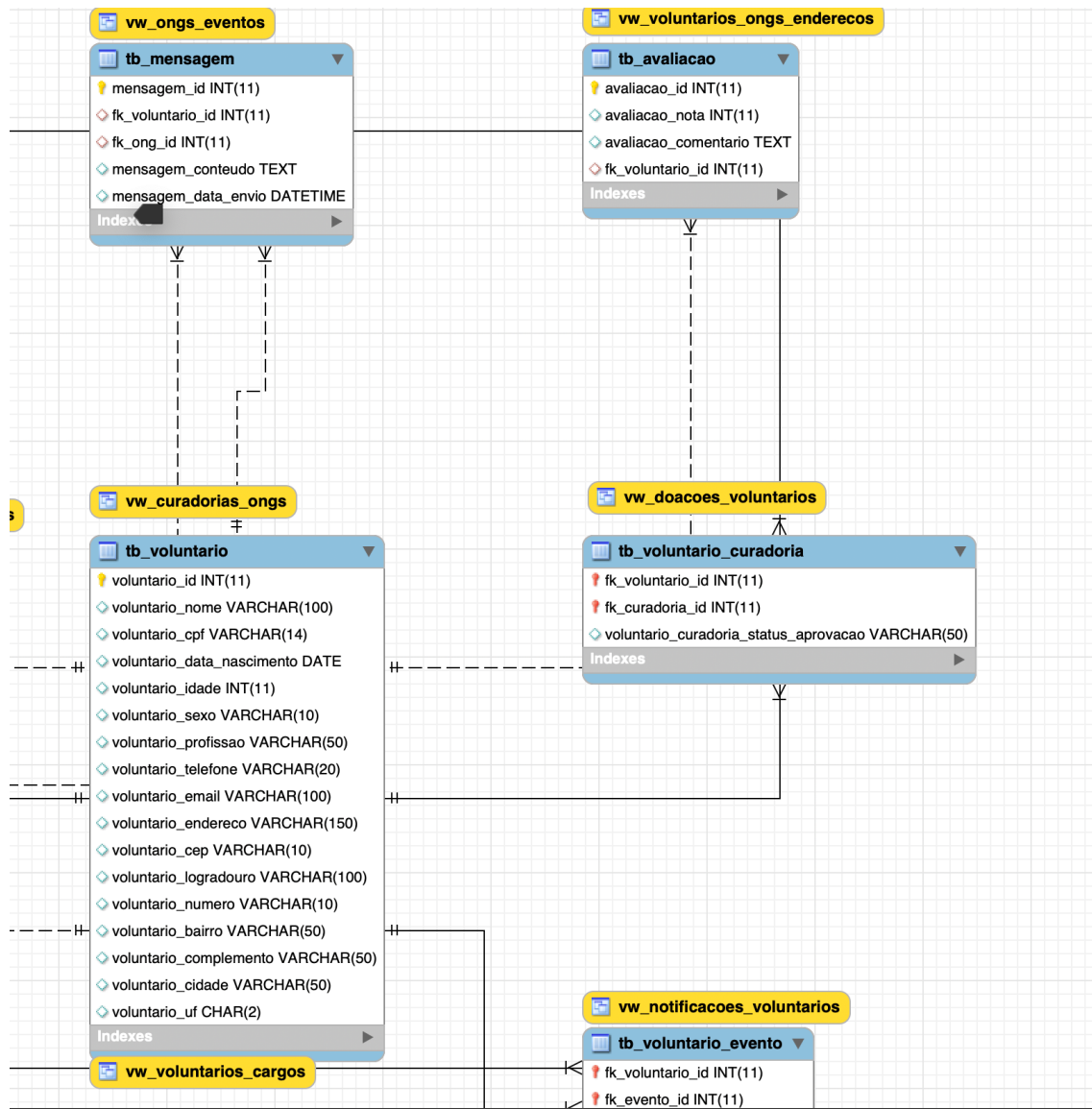
Um dos desafios mais difíceis foi garantir a criação de tabelas suficientes para garantir todos os requisitos. Isso inclui a criação das tabelas que relacionam 2 tabelas separadas, que só pôde ser decidida de verdade após criarmos todas as tabelas principais. Outro desafio foi a criação de views úteis e corretas para o nosso sistema. Isso foi um desafio, pois acreditávamos que poderíamos realizar a maior parte das consultas a partir de SELECTs, porém foi necessário a criação de views. Percebemos que as regras de negócio e as relações entre as tabelas exigiam consultas mais complexas, especialmente envolvendo múltiplos JOINS. O ultimo problema foi relacionado a população inicial do banco. Acreditávamos que com poucos INSERTs e VALUES teríamos o suficiente para fazer um teste real,

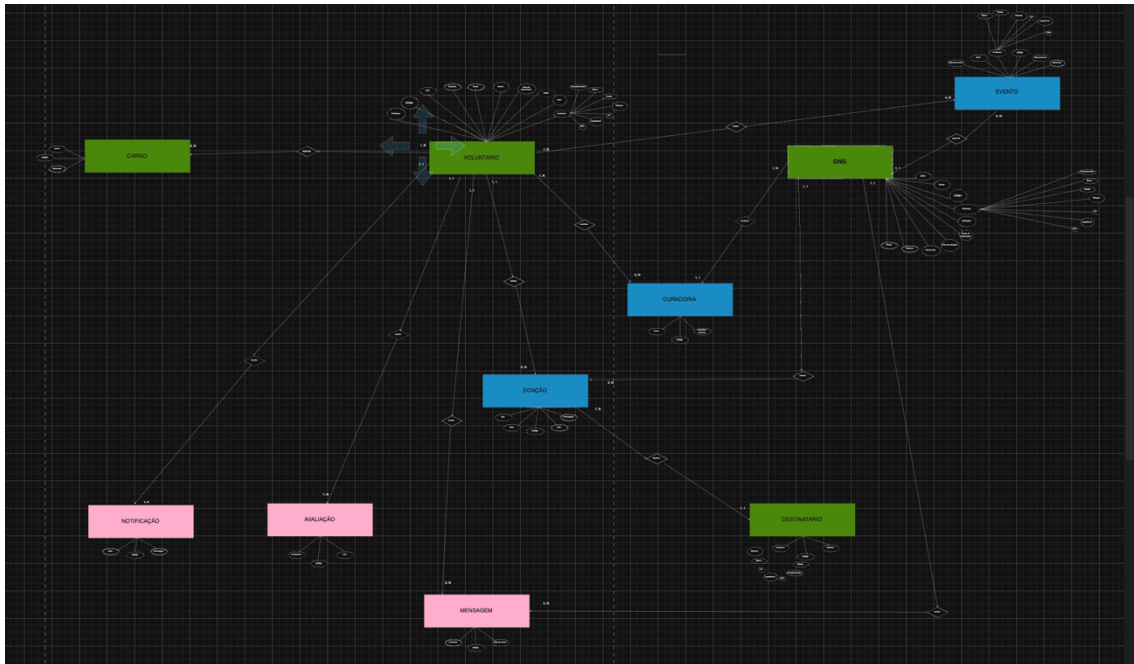
porém foi necessário a criação de mais INSERTs e maior população do banco para conseguirmos resultados de verdade.

Diagrama de Entidade e Relacionamento

Insira aqui a versão finalizada e ajustada do Diagrama de Entidade Relacionamento.







Lista de Requisitos e Comandos SQL

Aqui você irá inserir a lista de requisitos e seus respectivos comandos SQL conforme tabela abaixo:

Req 1	Cadastro de ONGs novas no sistema
SQL	<pre>INSERT INTO tb_ong (ong_nome, ong_email, ong_area_atuacao, ong_endereco, ong_cnpj, ong_telefone, ong_site, ong_nome_responsavel, ong_cep, ong_logradouro, ong_numero, ong_bairro, ong_complemento, ong_cidade, ong_uf) VALUES ('ONG Vida Verde', 'contato@vidaverde.org', 'Meio Ambiente', 'Rua das Árvores, 123, São Paulo', 99.888.777/0001-66', '(11) 97777-8888', 'www.vidaverde.org', 'João Verde', '01002-000', 'Rua das Árvores', '123', 'Jardim', '', 'São Paulo', 'SP');</pre>

Req 2	Lista de ONGs cadastradas no sistema
SQL	<pre>SELECT * FROM tb_ong;</pre>

Req 3	O sistema deve permitir o cadastro de voluntários novos no sistema
-------	--

SQL	INSERT INTO tb_voluntario (voluntario_nome, voluntario_email, voluntario_idade, voluntario_endereco, voluntario_cpf, voluntario_data_nascimento, voluntario_sexo, voluntario_profissao, voluntario_telefone, voluntario_cep, voluntario_logradouro, voluntario_numero, voluntario_bairro, voluntario_complemento, voluntario_cidade, voluntario_uf) VALUES ('Mariana Oliveira', 'mariana.oliveira@email.com', 28, 'Rua das Flores, 456, São Paulo', '777.888.999-00', 1997-04-15', 'Feminino', 'Designer', '(11) 98888-9999', '01003-000', 'Rua das Flores', '456', 'Centro', '', 'São Paulo', 'SP');

Req 4	Lista de voluntários cadastrados no sistema
SQL	CREATE VIEW vw_voluntarios_cadastrados AS SELECT v.voluntario_id, v.voluntario_nome, o.ong_id, o.ong_nome, o.ong_area_atuacao FROM tb_voluntario v JOIN tb_ong o ON v.voluntario_id = o.ong_id WHERE v.voluntario_id IS NOT NULL;

Req 5	Registro de novos eventos
SQL	INSERT INTO tb_evento (evento_titulo, evento_data_inicio, evento_data_termino, evento_endereco, evento_descricao, evento_cep, evento_logradouro, evento_numero, evento_bairro, evento_complemento, evento_cidade, evento_uf, fk_ong_id)

	VALUES ('Feira Verde', '2025-06-01', '2025-06-01', 'Rua das Árvores, 123, São Paulo', 'Feira para promover sustentabilidade', '01002-000', 'Rua das Árvores', '123', 'Jardim', '', 'São Paulo', 'SP', 1);
--	---

Req 6	Lista de eventos cadastrados
SQL	<pre>CREATE VIEW vw_eventos_cadastrados AS SELECT e.evento_id, e.evento_titulo, e.evento_data_inicio, e.evento_data_termino, o.ong_id, o.ong_nome FROM tb_evento e JOIN tb_ong o ON e.fk_ong_id = o.ong_id WHERE e.evento_id IS NOT NULL;</pre>

Req 7	Inscrição de voluntários em eventos
SQL	INSERT INTO tb_voluntario_evento (fk_voluntario_id, fk_evento_id) VALUES (3, 3);

Req 8	Lista de voluntários inscritos por eventos
SQL	<pre>CREATE VIEW vw_voluntarios_eventos AS SELECT v.voluntario_nome, e.evento_titulo, e.evento_data_inicio, e.evento_endereco FROM tb_voluntario v JOIN tb_voluntario_evento ve ON v.voluntario_id = ve.fk_voluntario_id JOIN tb_evento e ON ve.fk_evento_id = e.evento_id;</pre>

Req 9	Registro de doações dos voluntários para ONGs específicas
-------	---

SQL	INSERT INTO tb_doacao (doacao_tipo, doacao_valor, doacao_data, doacao_mensagem, fk_voluntario_id, fk_ong_id, fk_destinatario_id) VALUES ('Dinheiro', 100.00, '2025-05-28', 'Doação para projetos ambientais', 1, 1, 1);
-----	--

Req 10	Lista de doações recebidas para cada ONG, por voluntario
SQL	CREATE VIEW vw_doacoes_voluntarios AS SELECT v.voluntario_nome, d.doacao_tipo, d.doacao_valor, d.doacao_data, o.ong_nome, dest.destinatario_nome FROM tb_doacao d JOIN tb_voluntario v ON d.fk_voluntario_id = v.voluntario_id JOIN tb_ong o ON d.fk_ong_id = o.ong_id JOIN tb_destinatario dest ON d.fk_destinatario_id = dest.destinatario_id;

Req 11	Avaliação de voluntários por ONG
SQL	INSERT INTO tb_avaliacao (avaliacao_nota, avaliacao_comentario, fk_voluntario_id) VALUES (4, 'Ótimo desempenho no evento.', 1);

Req 12	Registro de novos curadores
SQL	INSERT INTO tb_curadoria (curadoria_nome, curadoria_requisitos_minimos, fk_ong_id) VALUES ('Curadoria de Sustentabilidade', 'Conhecimento em meio ambiente', 1);

Req 13	Lista de curadores
SQL	SELECT * FROM tb_curadoria;

Req 14	Envio e recebimento de mensagens, funcionando para ONGs e voluntários
--------	---

SQL	INSERT INTO tb_mensagem (fk_voluntario_id, fk_ong_id, mensagem_conteudo, mensagem_data_envio) VALUES (1, 1, 'Gostaria de participar do próximo evento.', '2025-05-28 16:54:00');
-----	---

Req 15	Consulta de mensagens trocadas entre ONGs e voluntários
SQL	CREATE VIEW vw_mensagens_ongs AS SELECT v.voluntario_nome, o.ong_nome, m.mensagem_conteudo, m.mensagem_data_envio FROM tb_mensagem m JOIN tb_voluntario v ON m.fk_voluntario_id = v.voluntario_id JOIN tb_ong o ON m.fk_ong_id = o.ong_id;

Req 16	Edição de perfil da ONG
SQL	UPDATE tb_ong SET ong_nome = 'ONG nova, Vida nova' WHERE ong_id = 1;

Req 16	Edição de perfil do voluntário
SQL	UPDATE tb_voluntario SET voluntario_email = 'emailnovo@email.com' WHERE ong_id = 4;

Req 17	Edição de perfil do evento
SQL	UPDATE tb_evento SET evento_endereco = 'Rua Augusto Zibarth, 1245, Curitiba' WHERE evento_id = 2;

Req 18	Edição de voluntários por evento
SQL	UPDATE tb_voluntario_evento SET fk_voluntario_id = 4 WHERE fk_evento_id = 2;

Req 19	Edição de avaliações
SQL	UPDATE tb_avaliacao SET avaliacao_nota = '2' WHERE fk_voluntario_id = 5;

Req 20	Edição de curadores
SQL	UPDATE tb_curadoria SET curadoria_nome = 'Curadoria de jogos eletrônicos ' WHERE fk_ong_id = 4;

Req 21	Exclusão de perfil da ONG
SQL	DELETE FROM tb_ong WHERE ong_id = 1;

Req 22	Edição de perfil do voluntário
SQL	DELETE FROM tb_voluntario WHERE voluntario_id = 5;

Req 23	Exclusão de perfil do evento
SQL	DELETE FROM tb_evento WHERE evento_id = 3;

Req 24	Exclusão de voluntários por evento
SQL	DELETE FROM tb_voluntario_evento WHERE fk_evento_id = 2;

Req 25	Exclusão de avaliações
SQL	DELETE FROM tb_avaliacao WHERE fk_voluntario_id = 5;

Req 26	Edição de curadores
SQL	DELETE FROM tb_curadoria WHERE fk_ong_id = 4

VIEWS, PROCEDURES E FUNCTIONS

Aqui você deve listar todas as Views, Procedures e Functions no Sistema,

```
CREATE VIEW
vw_voluntarios_eventos AS
SELECT
v.voluntario_nome,
e.evento_titulo,
e.evento_data_inicio,
e.evento_endereco
FROM
tb_voluntario v
JOIN tb_voluntario_evento ve ON v.voluntario_id = ve.fk_voluntario_id
```

JOIN tb_evento e ON ve.fk_evento_id = e.evento_id;

Exibe os voluntários cadastrados em cada evento, com nome, data e endereço do evento.

```
CREATE VIEW
    vw_doacoes_voluntarios AS
SELECT
    v.voluntario_nome,
    d.doacao_tipo,
    d.doacao_valor,
    d.doacao_data,
    o.ong_nome,
    dest.destinatario_nome
FROM
    tb_doacao d
    JOIN tb_voluntario v ON d.fk_voluntario_id = v.voluntario_id
    JOIN tb_ong o ON d.fk_ong_id = o.ong_id
    JOIN tb_destinatario dest ON d.fk_destinatario_id = dest.destinatario_id;
```

Mostra doações feitas por voluntários, incluindo tipo, valor, data, ONG e destinatário.

```
CREATE VIEW
    vw_avaliacoes_voluntarios AS
SELECT
    v.voluntario_nome,
    a.avaliacao_nota,
    a.avaliacao_comentario
FROM
    tb_avaliacao a
    JOIN tb_voluntario v ON a.fk_voluntario_id = v.voluntario_id;
```

Lista as avaliações dos voluntários com notas e comentários.

```
CREATE VIEW
    vw_voluntarios_cargos AS
SELECT
    v.voluntario_nome,
    c.cargo_nome
FROM
    tb_voluntario v
    JOIN tb_voluntario_cargo vc ON v.voluntario_id = vc.fk_voluntario_id
    JOIN tb_cargos c ON vc.fk_cargo_id = c.cargo_id;
```

Relaciona voluntários com os cargos que ocupam.

```

CREATE VIEW
    vw ONGS_eventos AS
SELECT
    o.ONG_nome,
    COUNT(e.evento_id) AS total_eventos
FROM
    tb_ONG o
    LEFT JOIN tb_evento e ON o.ONG_id = e.fk_ONG_id
GROUP BY
    o.ONG_nome;

```

Apresenta cada ONG com a quantidade de eventos que organizou.

```

CREATE VIEW
    vw_curadorias ONGS AS
SELECT
    c.curadoria_nome,
    o.ONG_nome,
    c.curadoria_requisitos_minimos
FROM
    tb_curadoria c
    JOIN tb_ONG o ON c.fk_ONG_id = o.ONG_id;

```

Mostra curadorias associadas às ONGs e seus requisitos mínimos.

```

CREATE VIEW
    vw_mensagens ONGS AS
SELECT
    v.voluntario_nome,
    o.ONG_nome,
    m.mensagem_conteudo,
    m.mensagem_data_envio
FROM
    tb_mensagem m
    JOIN tb_voluntario v ON m.fk_voluntario_id = v.voluntario_id
    JOIN tb_ONG o ON m.fk_ONG_id = o.ONG_id;

```

Lista mensagens enviadas por voluntários para ONGs, com data e conteúdo.

```

CREATE VIEW
    vw_notificacoes_voluntarios AS
SELECT
    v.voluntario_nome,
    n.notificacao_msg,
    n.notificacao_data
FROM
    tb_notificacao n
    JOIN tb_voluntario v ON n.fk_voluntario_id = v.voluntario_id;

```

Exibe notificações recebidas pelos voluntários, com data e conteúdo.

```

CREATE VIEW
    vw_voluntarios_ongs_enderecos AS
SELECT
    v.voluntario_nome,
    v.voluntario_endereco,
    o.ong_nome,
    o.ong_endereco
FROM
    tb_voluntario v
    JOIN tb_ong o ON v.voluntario_id = o.ong_id;

```

Relaciona voluntários e ONGs pelos seus nomes e endereços (possivelmente incorreta).

```

CREATE VIEW
    vw_aprovado_pendentes_voluntarios_curadorias AS
SELECT
    v.voluntario_nome,
    c.curadoria_nome,
    vc.voluntario_curadoria_status_aprovacao
FROM
    tb_voluntario_curadoria vc
    JOIN tb_voluntario v ON vc.fk_voluntario_id = v.voluntario_id
    JOIN tb_curadoria c ON vc.fk_curadoria_id = c.curadoria_id
WHERE
    vc.voluntario_curadoria_status_aprovacao IN ('Aprovado', 'Pendente');

```

Lista o status de aprovação (Aprovado/Pendente) dos voluntários nas curadorias.

```

CREATE VIEW
    vw_voluntarios_cadastrados AS
SELECT
    v.voluntario_id,
    v.voluntario_nome,
    o.ong_id,
    o.ong_nome,
    o.ong_area_atuacao
FROM
    tb_voluntario v
    JOIN tb_ong o ON v.voluntario_id = o.ong_id
WHERE
    v.voluntario_id IS NOT NULL;

```

Exibe voluntários cadastrados e a ONG à qual estão associados (possível erro de JOIN).

```

CREATE VIEW
    vw_eventos_cadastrados AS
SELECT
    e.evento_id,
    e.evento_titulo,
    e.evento_data_inicio,
    e.evento_data_termino,
    o.ong_id,
    o.ong_nome
FROM
    tb_evento e
    JOIN tb_ong o ON e.fk_ong_id = o.ong_id
WHERE
    e.evento_id IS NOT NULL;

```

Mostra eventos cadastrados com seus dados e a ONG responsável.