



UNIVERSIDADE
FEDERAL DO CEARÁ

Fundamentos de Banco de Dados
Relatório de Projeto de Banco de Dados

Discentes:

Abraão Levi de Andrade Pessoa Vitoriano - 548324

João Guilherme Pinto de Souza do Nascimento - 571799

José Nilson Bernardo Nunes Filho - 567157

Matheus Narcizio Cavalcante de Macedo - 494693

Docente: Livia Almada Cruz

Título: Aplicativo de Mapeamento de Rotas Acessíveis

QUIXADÁ 2024.2

Introdução

O Aplicativo de Mapeamento de Rotas Acessíveis é uma solução projetada para promover a inclusão e a mobilidade urbana para pessoas com deficiência, idosos e outros grupos que enfrentam desafios de acessibilidade no deslocamento diário. A aplicação visa facilitar o planejamento de trajetos acessíveis, fornecendo informações detalhadas sobre rotas adaptadas, condições das vias e a disponibilidade de recursos como rampas, elevadores e transporte público acessível.

Objetivos da Aplicação

- **Promover a Inclusão Social:** Fornecer informações que permitam a todos, independentemente de suas limitações físicas, transitar de forma segura e independente.
- **Facilitar o Planejamento de Rotas:** Auxiliar usuários a identificarem trajetos que melhor atendam às suas necessidades de acessibilidade e mobilidade.
- **Oferecer Feedback Colaborativo:** Permitir que os usuários avaliem e contribuam com informações sobre as condições das rotas, garantindo um mapeamento dinâmico e atualizado.
- **Incentivar Melhorias na Infraestrutura:** Fornecer dados que podem ser utilizados por autoridades locais para identificar áreas que necessitam de investimentos em acessibilidade.
- **Aumentar a Qualidade de Vida:** Reduzir barreiras no deslocamento diário, proporcionando mais autonomia e qualidade de vida aos usuários.

Requisitos Funcionais

1. **Cadastro de Rotas:** Permitir que o usuário insira, edite, remova e liste rotas acessíveis. Cada rota deve conter o nome, pontos de partida e destino, descrição (ex.: presença de rampas, semáforos sonoros, e condições das calçadas).
2. **Consulta de Rotas por Critérios:** Permitir que o usuário pesquise rotas baseando-se em critérios como: nível de acessibilidade, distância, tempo estimado, ou tipo de transporte utilizado.
3. **Relatório de Utilização de Rotas:** Gerar um relatório que liste as rotas mais utilizadas em um período específico, incluindo informações sobre o nível de acessibilidade e feedback dos usuários.
4. **Feedback de Usuários sobre Rotas:** Permitir que os usuários avaliem rotas, dando notas e comentários sobre a acessibilidade, segurança e facilidade de uso.
5. **Notificações de Alterações nas Rotas:** Enviar notificações para os usuários quando houver alterações em uma rota que eles costumam utilizar.
6. **Cadastro de Pontos de Interesse:** Permitir que o usuário insira, edite, remova e liste pontos de interesse próximos às rotas.

7. **Consulta de Pontos de Interesse por Categorias:** Permitir que o usuário pesquise pontos de interesse baseando-se em categorias como: saúde, lazer, educação, transporte público.
8. **Histórico de Utilização de Rotas:** Permitir que os usuários visualizem o histórico de rotas utilizadas, com detalhes como data, hora, e o feedback dado (se houver).
9. **Sistema de Recomendação de Rotas:** Recomendar rotas com base no histórico de uso e nas preferências salvas pelos usuários.
10. **Monitoramento em Tempo Real:** Exibir informações em tempo real sobre o status das rotas, como trânsito, bloqueios temporários e condições climáticas.
11. **Modo Offline:** Permitir que os usuários salvem rotas e pontos de interesse para consulta offline.
12. **Gerenciamento de Favoritos:** Permitir que os usuários marquem rotas e pontos de interesse como favoritos, organizando-os em categorias personalizáveis.
13. **Cadastro do Usuário:** Permitir que o usuário realize o cadastro no sistema informando nome, idade, sexo, telefone de contato e CPF.
14. **Suporte por Chat:** Disponibilizar um sistema de suporte por chat para que o usuário possa tirar dúvidas.
15. **Solicitar Criação de Rotas Personalizadas:** Permitir que o usuário solicite a criação de rotas personalizadas com base em suas necessidades específicas.
16. **Cadastro de Transportes Públicos:** Permitir o cadastro de linhas de transporte público associadas às rotas.
17. **Consulta de Transportes Públicos:** Permitir que os usuários pesquisem linhas de transporte público associadas às rotas.

Desenvolvimento

Entidades e Atributos

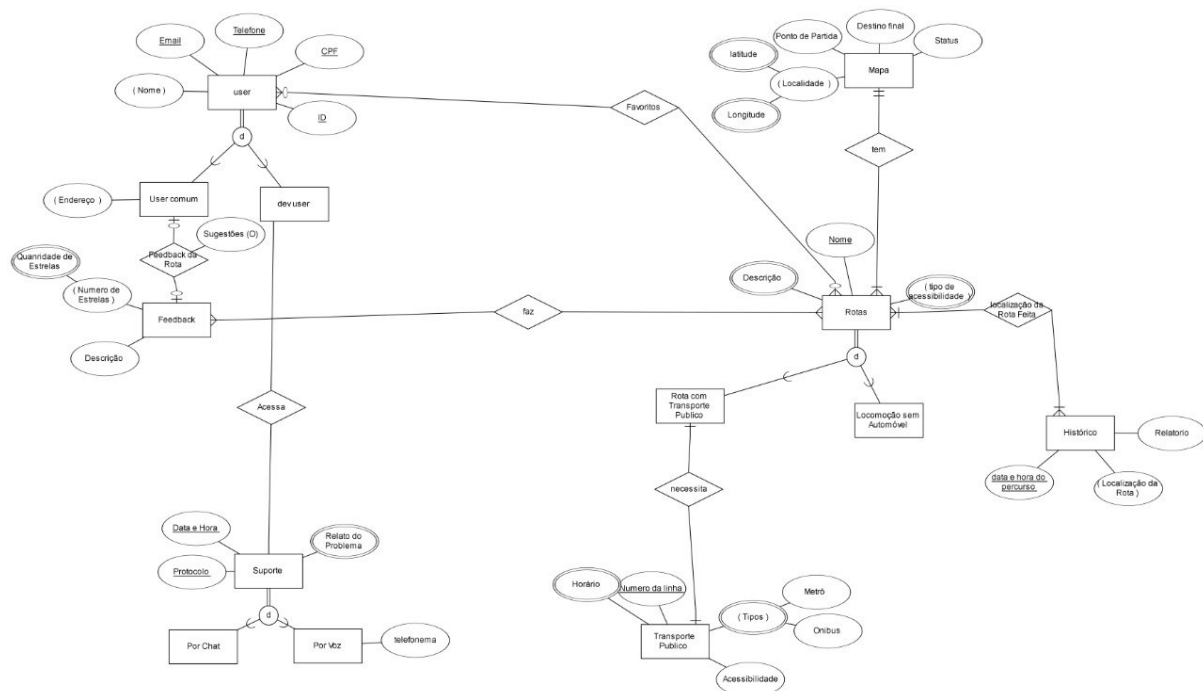
- **Entidade User:** Email, Telefone, Nome, Senha, Tipo de Acessibilidade, Preferência de Acessibilidade, Localização Atual, ID.
- **Entidade Transporte Público:** Horário, Tipo, Acessibilidade, Nome, ID.
- **Entidade Rota:** Nome, Descrição, Tipo de Acessibilidade, ID, ID Usuário, ID Destino, Distância.
- **Entidade Histórico:** ID Usuário, ID, ID Ponto Rota, Data e Hora do Percuro.
- **Entidade Localização Usuário:** ID, ID Usuário, Coordenadas, Data e Hora.
- **Entidade Avaliação:** ID, ID Usuário, ID Rota, Nota, Comentário, Data.

- **Entidade Rota Transporte:** ID Rota, ID Transporte.
- **Entidade Ponto de Interesse:** ID, Nome, Tipo, Localidade, Disponibilidade.
- **Entidade Relatório de Problema:** ID, ID Usuário, ID Ponto Rota, Descrição, Data, Status, Prioridade.
- **Entidade Rota Ponto:** ID Rota, ID Ponto.

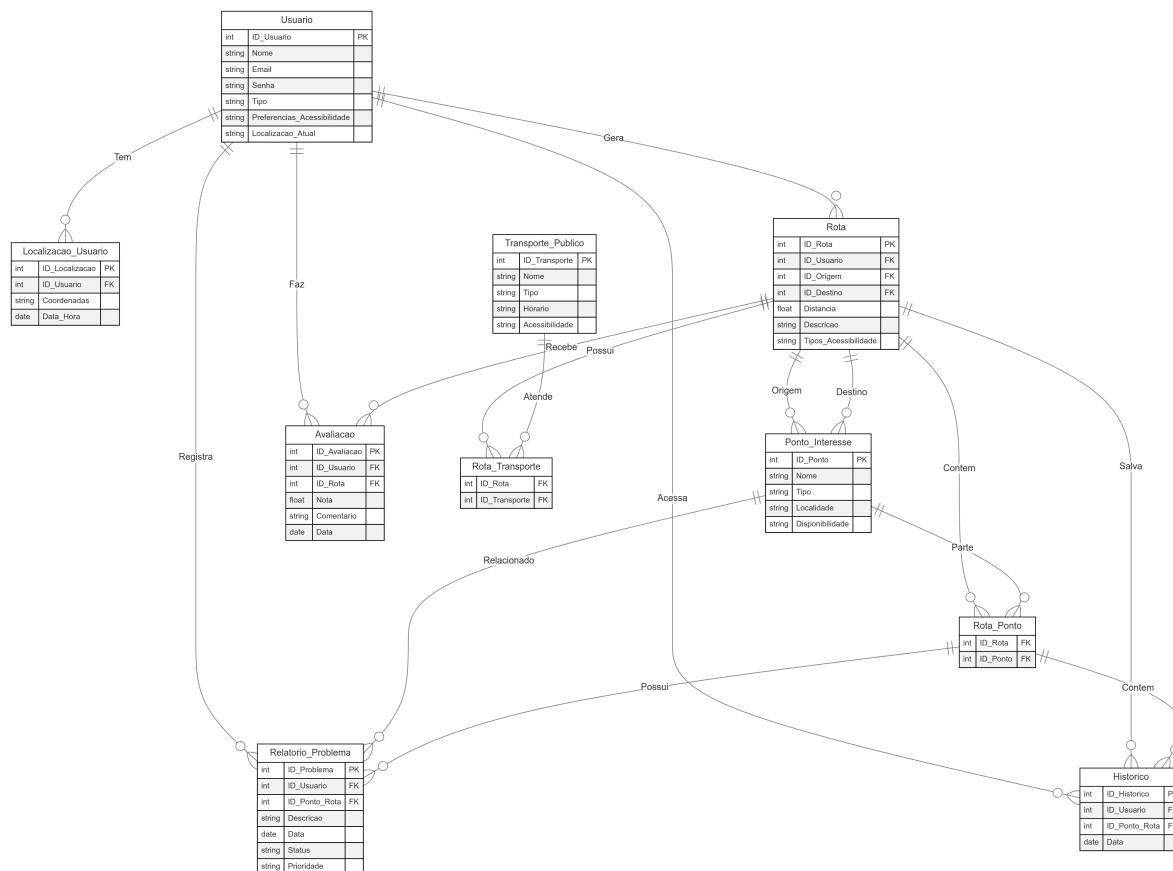
Relacionamentos

- **Usuário** tem uma **localização** (Relacionamento N:1).
- **Usuário** faz uma **Avaliacao da Rota**. (Relacionameto 1:N).
- **Usuário** gera uma **Rota** (Relacionamento 1:N).
- **Usuário** registra um **Relatório de Problemas** (Relacionamento 1:N).
- **Rota** recebe **Avaliações** (Relacionamento 1:N).
- **Rota** possui **Transportes** (Relacionamento N:M).
- **Transporte Público** atende **Rota Transporte** (Relacionamento 1:N).
- **Relatório de Problemas** tem salvo o **Usuário**, **Ponto Rota** (Relacionamento 1:N).

Diagrama ER



Esquema Relacional



Chaves Primárias e Estrangeiras das Tabelas

- **Usuário:** Uma chave primária(ID).
- **Localização Usuário:** Uma chave primária(ID) e uma chave estrangeira para Usuário(ID usuário).
- **Rota:** Uma chave primária(ID), uma chave estrangeira para Usuário(ID usuario), e uma chave estrangeira para Ponto de Interesse(ID destino).
- **Avaliação:** Uma chave primária(ID), uma chave estrangeira para Usuário(ID usuario), e uma chave estrangeira para Rota(ID rota).
- **Transporte Público:** Uma chave primária(ID).
- **Rota Transporte:** Uma chave estrangeira para Rota(ID rota), e uma chave estrangeira para Transporte Público(ID transporte).
- **Ponto de Interesse:** Uma chave primária(ID).
- **Rota Ponto:** Uma chave estrangeira para Rota(ID rota), e uma chave estrangeira para Ponto de Interesse(ID ponto).

- **Relatório de Problema:** Uma chave primária(ID), uma chave estrangeira para Usuário(ID usuario), e uma chave estrangeira para Ponto Rota(ID ponto rota).
- **Histórico:** Uma chave primária(ID), uma chave estrangeira para Usuário(ID usuário), e uma chave estrangeira para Ponto Rota(ID ponto rota).

Tabelas e suas Tuplas

- **Usuário:** Guarda todas as informações dos usuários, inclusive se é um usuário do tipo comum ou administrador.
- **Localização Usuário:** Guarda todas as informações sobre a localização do usuário, como suas coordenadas.
- **Avaliação:** Contém todas as avaliações dos usuários em relação às rotas que eles utilizaram, com um sistema de notas de 1 à 5.
- **Transporte Público:** Contém todas as informações sobre o transporte público que será utilizado na rota, inclusive o tipo de acessibilidade.
- **Rota Transporte:** Guarda a relação entre rotas e os transportes disponíveis.
- **Rota:** Guarda uma rota que pode incluir vários pontos de interesse, contendo o tipo de acessibilidade que aquela rota oferece e sua descrição para melhor segurança do usuário.
- **Ponto de Interesse:** Contém o destino que o usuário deseja alcançar na rota, podendo ser um parque, cinema, ou qualquer lugar que seja um ponto de referência.
- **Rota Ponto:** Armazena a relação entre a rota e os pontos de interesse que ela possui, ou seja, os destinos.
- **Histórico:** Contém informações sobre todas as rotas que o usuário percorreu.
- **Relatório de Problema:** Registra todos os problemas que os usuários tiveram na rota e nos pontos de interesse, tendo as informações como status daquele ponto ou rota e a prioridade do problema que ele tem, ou seja, que o administrador tem que considerar.

Script da implementação

```
CREATE TABLE usuario (
    id_usuario SERIAL PRIMARY KEY,
    nome VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    senha VARCHAR(100) NOT NULL,
    tipo VARCHAR(100),
    preferencias_acessibilidade VARCHAR(100),
    localizacao_atual VARCHAR(100)
);
```

```

CREATE TABLE localizacao_usuario (
    id_localizacao SERIAL PRIMARY KEY,
    coordenadas VARCHAR(100),
    data_hora DATE,
    id_usuario INT,
    FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario)
);

CREATE TABLE transporte_publico (
    id_transporte SERIAL PRIMARY KEY,
    nome VARCHAR(100),
    tipo VARCHAR(100),
    horario TIME,
    acessibilidade VARCHAR(100)
);

CREATE TABLE ponto_interesse (
    id_ponto SERIAL PRIMARY KEY,
    nome VARCHAR(100),
    tipo VARCHAR(100),
    locabilidade VARCHAR(100),
    disponibilidade VARCHAR(100)
);

CREATE TABLE rota (
    id_rota SERIAL PRIMARY KEY,
    distancia FLOAT,
    descricao VARCHAR(100),
    tipo_acessibilidade VARCHAR(100),
    id_usuario INT,
    FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario),
    id_destino INT,
    FOREIGN KEY (id_destino) REFERENCES ponto_interesse(id_ponto)
);

CREATE TABLE rota_transporte (
    id_rota INT,
    FOREIGN KEY (id_rota) REFERENCES rota(id_rota),
    id_transporte INT,
    FOREIGN KEY (id_transporte) REFERENCES
        ↪ transporte_publico(id_transporte)
);

CREATE TABLE avaliacao (
    id_avaliacao SERIAL PRIMARY KEY,
    nota FLOAT,
    comentario VARCHAR(100),
    data DATE,

```

```

        id_usuario INT,
        FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario),
        id_rota INT,
        FOREIGN KEY (id_rota) REFERENCES rota(id_rota)
    );

CREATE TABLE rota_ponto (
    id_rota INT,
    FOREIGN KEY (id_rota) REFERENCES rota(id_rota),
    id_ponto INT,
    FOREIGN KEY (id_ponto) REFERENCES ponto_interesse(id_ponto)
);

CREATE TABLE relatorio_problema (
    id_problema SERIAL PRIMARY KEY,
    descricao VARCHAR(100),
    data DATE,
    status VARCHAR(100),
    prioridade VARCHAR(100),
    id_usuario INT,
    FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario),
    id_ponto_rota INT,
    FOREIGN KEY (id_ponto_rota) REFERENCES ponto_interesse(id_ponto)
);

CREATE TABLE historico (
    id_historico SERIAL PRIMARY KEY,
    data DATE NOT NULL,
    id_usuario INT NOT NULL,
    id_ponto_rota INT NOT NULL,
    FOREIGN KEY (id_usuario) REFERENCES usuario(id_usuario),
    FOREIGN KEY (id_ponto_rota) REFERENCES ponto_interesse(id_ponto)
);

```

Popularizando a tabela

```

-- Inserção de dados na tabela `usuario`
INSERT INTO usuario (id_usuario, nome, email, senha, tipo,
    ↪ preferencias_acessibilidade, localizacao_atual)
VALUES
    (1, 'Lucas Almeida', 'lucas@xyz.com', '54321', 'usuario comum',
    ↪ 'nenhuma', 'Fortaleza'),
    (2, 'Ana Beatriz', 'ana@abc.com', '98765', 'usuario comum',
    ↪ 'nenhuma', 'Recife'),
    (3, 'Gabriel Souza', 'gabriel@def.com', '45678', 'usuario comum',
    ↪ 'nenhuma', 'Salvador'),
    (4, 'Mariana Lima', 'mariana@ghi.com', '11223', 'usuario comum',
    ↪ 'nenhuma', 'Curitiba'),

```



```

(5, 'Felipe Costa', 'felipe@jkl.com', '33445', 'usuario comum',
  → 'nenhuma', 'São Paulo'),
(6, 'Juliana Mendes', 'juliana@mno.com', '66789', 'usuario comum',
  → 'nenhuma', 'Rio de Janeiro'),
(7, 'Rafael Santos', 'rafael@pqr.com', '99887', 'usuario comum',
  → 'nenhuma', 'Belo Horizonte'),
(8, 'Camila Oliveira', 'camila@stu.com', '22446', 'usuario comum',
  → 'nenhuma', 'Porto Alegre'),
(9, 'Bruno Ferreira', 'bruno@vwx.com', '77889', 'usuario comum',
  → 'nenhuma', 'Manaus'),
(10, 'Matheus Narcizio', 'matheus@fbd.com', '12345', 'adm',
  → 'nenhuma', 'Quixada');

-- Inserção de dados na tabela `localizacao_usuario`
INSERT INTO localizacao_usuario (id_localizacao, coordenadas, data_hora,
  → id_usuario)
VALUES
  (1, '(-23.550520, -46.633308)', '2024-02-01', 4),
  (2, '(-22.906847, -43.172897)', '2024-02-02', 6),
  (3, '(-19.916681, -43.934493)', '2024-02-03', 2),
  (4, '(-15.826691, -47.921822)', '2024-02-04', 3),
  (5, '(-12.971399, -38.501632)', '2024-02-05', 5),
  (6, '(-8.047562, -34.877018)', '2024-02-06', 1),
  (7, '(-3.731862, -38.526669)', '2024-02-07', 8),
  (8, '(-30.034647, -51.217658)', '2024-02-08', 7),
  (9, '(-1.455754, -48.490179)', '2024-02-09', 9),
  (10, '(-25.428954, -49.267137)', '2024-02-10', 10);

-- Inserção de dados na tabela `transporte_publico`
INSERT INTO transporte_publico (id_transporte, nome, tipo, horario,
  → acessibilidade)
VALUES
  (1, 'Ônibus 101', 'Ônibus', '06:30:00', 'Acessível para
    → cadeirantes'),
  (2, 'Metrô Linha 1', 'Metrô', '07:15:00', 'Elevadores disponíveis'),
  (3, 'BRT Expresso', 'BRT', '08:00:00', 'Rampa de acesso'),
  (4, 'VLT Centro', 'VLT', '09:45:00', 'Espaço para cadeirantes'),
  (5, 'Ônibus 202', 'Ônibus', '11:20:00', 'Acessível para cegos e
    → surdos'),
  (6, 'Metrô Linha 2', 'Metrô', '13:10:00', 'Plataforma adaptada'),
  (7, 'Bonde Histórico', 'Bonde', '15:30:00', 'Sem acessibilidade'),
  (8, 'Ferry Boat', 'Ferry', '17:00:00', 'Acessível para cadeirantes e
    → idosos'),
  (9, 'Ônibus Noturno', 'Ônibus', '22:45:00', 'Acessível para todos'),
  (10, 'Metrô Linha 3', 'Metrô', '23:55:00', 'Elevadores e rampas
    → disponíveis');

-- Inserção de dados na tabela `ponto_interesse`

```

```

INSERT INTO ponto_interesse (id_ponto, nome, tipo, locabilidade,
↪ disponibilidade)
VALUES
(1, 'Parque Central', 'Parque', 'Área urbana', '24 horas'),
(2, 'Museu de Arte', 'Museu', 'Centro da cidade', 'Ter-Sáb 10:00 -
↪ 18:00'),
(3, 'Biblioteca Municipal', 'Biblioteca', 'Bairro universitário',
↪ 'Seg-Sex 08:00 - 20:00'),
(4, 'Shopping Plaza', 'Shopping', 'Zona comercial', 'Diariamente
↪ 10:00 - 22:00'),
(5, 'Estádio Nacional', 'Estádio', 'Região metropolitana', 'Eventos
↪ programados'),
(6, 'Praia do Sol', 'Praia', 'Litoral', 'Livre acesso'),
(7, 'Teatro Municipal', 'Teatro', 'Centro histórico', 'Conforme
↪ programação'),
(8, 'Zoológico da Cidade', 'Zoológico', 'Área verde', 'Qua-Dom 09:00
↪ - 17:00'),
(9, 'Centro de Convenções', 'Centro de Eventos', 'Região central',
↪ 'Conforme eventos'),
(10, 'Mirante da Serra', 'Mirante', 'Área montanhosa', '06:00 -
↪ 18:00');

```

-- Inserção de dados na tabela `rota`

```

INSERT INTO rota (id_rota, distancia, descricao, tipo_acessibilidade,
↪ id_usuario, id_destino)
VALUES
(1, 2.5, 'Caminho até o parque', 'Rampa para cadeirantes', 1, 1),
(2, 5.2, 'Rota até o museu', 'Elevadores disponíveis', 2, 2),
(3, 3.8, 'Trilha para a biblioteca', 'Acesso pavimentado', 3, 3),
(4, 1.5, 'Rota curta para o shopping', 'Totalmente acessível', 4, 4),
(5, 7.3, 'Percurso até o estádio', 'Sem acessibilidade', 5, 5),
(6, 4.0, 'Caminho para a praia', 'Rampa de acesso', 6, 6),
(7, 6.5, 'Trilha até o teatro', 'Apenas escadas', 7, 7),
(8, 3.2, 'Rota até o zoológico', 'Rampas e piso tátil', 8, 8),
(9, 5.7, 'Percurso até o centro de convenções', 'Acessível para
↪ todos', 9, 9),
(10, 8.0, 'Trilha até o mirante', 'Parcialmente acessível', 10, 10);

```

-- Inserção de dados na tabela `rota_transporte`

```

INSERT INTO rota_transporte (id_rota, id_transporte)
VALUES
(1, 1),
(2, 2),
(3, 3),
(4, 4),
(5, 5),
(6, 6),
(7, 7),

```

```

(8, 8),
(9, 9),
(10, 10);

-- Inserção de dados na tabela `avaliacao`
INSERT INTO avaliacao (id_avaliacao, nota, comentario, data, id_usuario,
↳ id_rota)
VALUES
(1, 4.5, 'Rota muito boa, bem sinalizada.', '2024-02-01', 1, 1),
(2, 3.0, 'Acessibilidade poderia ser melhor.', '2024-02-02', 2, 2),
(3, 5.0, 'Excelente percurso, sem problemas.', '2024-02-03', 3, 3),
(4, 2.5, 'Houve algumas dificuldades na rota.', '2024-02-04', 4, 4),
(5, 4.0, 'A rota foi tranquila, mas um pouco longa.', '2024-02-05',
↳ 5, 5),
(6, 3.5, 'Rota razoável, mas poderia melhorar.', '2024-02-06', 6, 6),
(7, 5.0, 'A melhor rota até agora, super tranquila!', '2024-02-07',
↳ 7, 7),
(8, 2.0, 'A rota foi um pouco difícil de navegar.', '2024-02-08', 8,
↳ 8),
(9, 4.2, 'Boa rota, mas com alguns obstáculos.', '2024-02-09', 9, 9),
(10, 3.8, 'Acessibilidade boa, mas poderia ter mais pontos de
↳ descanso.', '2024-02-10', 10, 10);

-- Inserção de dados na tabela `rota_ponto`
INSERT INTO rota_ponto (id_rota, id_ponto)
VALUES
(1, 1),
(1, 2),
(2, 3),
(2, 4),
(3, 5),
(3, 6),
(4, 7),
(4, 8),
(5, 9),
(5, 10);

-- Inserção de dados na tabela `relatorio_problema`
INSERT INTO relatorio_problema (id_problema, descricao, data, status,
↳ prioridade, id_usuario, id_ponto_rota)
VALUES
(1, 'Lixo acumulado na entrada do parque', '2024-02-01', 'pendente',
↳ 'alta', 1, 1),
(2, 'Problema na sinalização de trânsito no museu', '2024-02-02', 'em
↳ andamento', 'média', 2, 2),
(3, 'Falta de acessibilidade no zoológico', '2024-02-03',
↳ 'resolvido', 'alta', 3, 3),

```

```

(4, 'Escada quebrada no shopping', '2024-02-04', 'pendente', 'média',
↪ 4, 4),
(5, 'Iluminação insuficiente no estádio', '2024-02-05', 'em
↪ andamento', 'baixa', 5, 5),
(6, 'Vias esburacadas no centro de convenções', '2024-02-06',
↪ 'pendente', 'alta', 6, 6),
(7, 'Acessibilidade inadequada na praia', '2024-02-07', 'resolvido',
↪ 'alta', 7, 7),
(8, 'Problema de segurança no teatro', '2024-02-08', 'pendente',
↪ 'média', 8, 8),
(9, 'Falta de limpeza no mirante', '2024-02-09', 'em andamento',
↪ 'baixa', 9, 9),
(10, 'Problema de sinalização na biblioteca', '2024-02-10',
↪ 'resolvido', 'baixa', 10, 10);

-- Inserção de dados na tabela `historico`
INSERT INTO historico (id_historico, data, id_usuario, id_ponto_rota)
VALUES
  (1, '2024-02-01', 1, 1),
  (2, '2024-02-02', 2, 2),
  (3, '2024-02-03', 3, 3),
  (4, '2024-02-04', 4, 4),
  (5, '2024-02-05', 5, 5),
  (6, '2024-02-06', 6, 6),
  (7, '2024-02-07', 7, 7),
  (8, '2024-02-08', 8, 8),
  (9, '2024-02-09', 9, 9),
  (10, '2024-02-10', 10, 10);

```

Implementação do CRUD em python

Nessa seção, é mostrada as alterações que foram feitas nas tabelas que foram entregue anteriormente, e as telas feitas com a biblioteca **panel** do **python** que permite a criação de interfaces interativas de maneira simples e prática.

Alterações da entrega anterior

Na parte da popularização das tabelas, foram inseridas novas tuplas na tabela **rota**, para permitir que tenham avaliações de rotas que não foram ainda adicionadas.

```

INSERT INTO rota (distancia, descricao, tipo_acessibilidade,
↪ id_usuario, id_destino)
VALUES
  (3.0, 'Trilha até o zoológico', 'Acesso pavimentado', 2, 8),
  (5.0, 'Caminho até o museu', 'Rampas para cadeirantes', 5, 2),
  (2.0, 'Percuso até o teatro', 'Parcialmente acessível', 1, 7);

```

Crud Pontos de Interesse Acessíveis (CRUD)

- **Tela de Cadastro de Pontos de Interesse** Permite que o usuário insira um novo ponto de interesse acessível. Contém campos para nome, localização, tipo de acessibilidade (rampa, elevador, banheiro adaptado etc.), horário de funcionamento e localidade. Possui um botão para salvar as informações no banco de dados.
- **Tela de Listagem e Consulta**
Exibe todos os pontos de interesse cadastrados em formato de tabela. Possui filtros para buscar locais por tipo de acessibilidade, horário de funcionamento e localização. Inclui uma barra de pesquisa para encontrar pontos específicos.
- **Tela de Edição de Pontos de Interesse**
Permite a modificação das informações de um ponto de interesse existente. O usuário pode alterar qualquer campo e salvar as mudanças. Mostra a tabela organizada pelo `id_ponto` ao final
- **Tela de Exclusão de Pontos de Interesse**
Exclui uma tupla da tabela, utilizando o nome para a remoção. Mostra a tabela organizada pelo `id_ponto` ao final.

Crud Rotas Acessíveis (CRUD)

- **Tela de Cadastro de Rotas**
Permite que o usuário insira uma nova rota acessível. Contém campos para distância, descrição da rota, tipo de acessibilidade (piso tátil, rampa, elevador etc.) e usuário responsável pelo cadastro. Possui um botão para salvar a rota no banco de dados.
- **Tela de Listagem e Consulta de Rotas**
Exibe todas as rotas cadastradas em formato de tabela. Inclui filtros para buscar rotas por tipo de acessibilidade, distância e localização. Possui uma barra de pesquisa para facilitar a busca por rotas específicas.
- **Tela de Edição de Rotas**
Permite a modificação das informações de uma rota existente. O usuário pode atualizar qualquer campo, como a distância ou o tipo de acessibilidade. Inclui um botão de confirmação para garantir que as alterações sejam salvas corretamente.
- **Tela de Exclusão de Rotas**
Exibe uma lista das rotas cadastradas com a opção de excluir um registro. Inclui um alerta de confirmação antes de excluir definitivamente a rota do banco de dados.

Crud Avaliação de usuário(CRUD)

- **Consulta de Avaliações**
Permite que o usuário filtre e visualize avaliações cadastradas. Contém campos para buscar por nome do usuário, nome da rota, comentário, nota e data. Exibe os resultados em uma tabela interativa.

- **Cadastro de Avaliações**

Permite que o usuário cadastre uma nova avaliação para uma rota específica. Valida se o usuário e a rota existem antes de salvar. Possui um botão para salvar a avaliação no banco de dados.

- **Edição de Avaliações**

Permite a modificação das informações de uma avaliação existente. O usuário pode atualizar o comentário e a nota atribuída à rota. Inclui um botão de confirmação para garantir que as alterações sejam salvas corretamente.

- **Exclusão de Avaliações**

Exibe uma lista das avaliações cadastradas com a opção de excluir um registro. Inclui um alerta de confirmação antes de excluir definitivamente a avaliação do banco de dados.

CRUD Usuário(CRUD)

- **Conexão com o Banco de Dados** Utiliza SQLAlchemy para conectar-se a um banco PostgreSQL. Define a URI do banco e cria um mecanismo de conexão. Configura sessões para interagir com o banco de dados.

- **Definição da Tabela de Usuários**

Cria a classe `Usuario` representando a tabela `usuario`. Define colunas como ID, nome, email, senha, tipo, preferências de acessibilidade e localização atual. Garante que o email seja único e que certos campos sejam obrigatórios.

- **Operações no Banco de Dados**

Implementa a função `executar_operacao()` para gerenciar transações. Inclui funções para criar, listar, atualizar e excluir usuários. Retorna feedback ao usuário sobre o sucesso ou falha das operações.

- **Interface com Panel**

Utiliza a extensão `panel` para criar uma interface interativa. Atualiza automaticamente a tabela de usuários ao realizar operações.

- **Componentes da Interface**

Inclui campos de entrada para nome, email, senha, tipo, preferências e localização. Possui botões para criar, atualizar e excluir usuários, com feedback visual. Exibe a lista de usuários em formato de tabela interativa.

- **Callbacks e Interatividade**

Define funções para lidar com eventos de clique nos botões. Atualiza automaticamente a interface conforme as operações são executadas. Exibe mensagens de sucesso ou erro conforme a operação realizada.

- **Execução da Interface**

Organiza os componentes em um layout responsivo. Usa `layout.serveable()` para disponibilizar a interface para acesso via navegador.

Delegação das Atividades

Atividade	Descrição	Responsável
CRUD Usuário	Criou o CRUD para a tabela Usuário	João Guilherme
CRUD Avaliação	Criou o CRUD para a tabela Avaliação	Jose Nilson Bernardo
CRUD ponto de interesse	Criou o CRUD para a tabela Pontos de Interesse	Matheus Narcizio
CRUD Rotas	Criou o CRUD para a tabela Rotas	Abraão Levi