

Estrutura Banco de Dados NextWork, utilizando PostgreSQL:

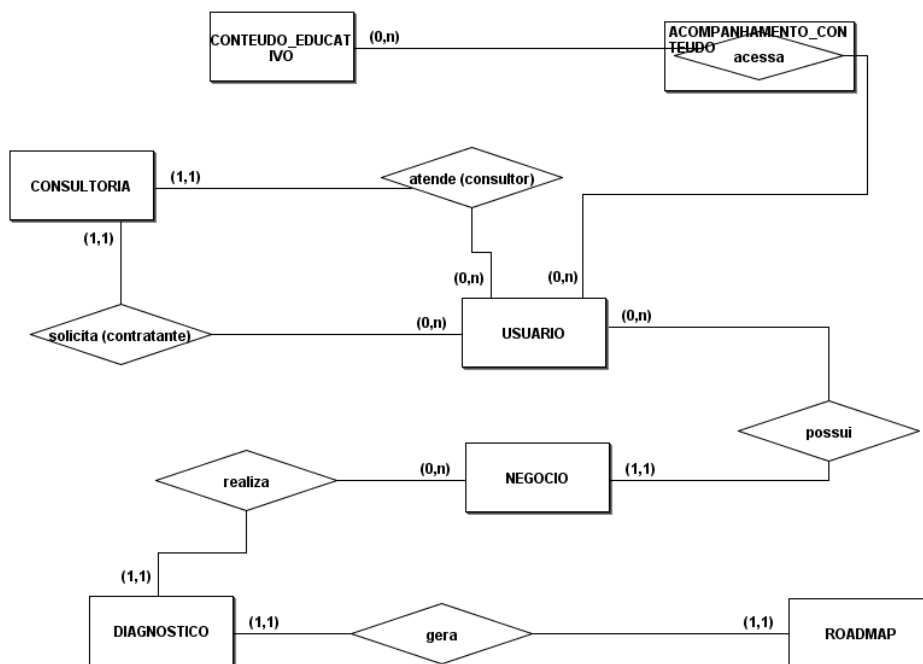
Entidade	Propósito	Atributos Chave
USUARIO	Armazena dados dos usuários (comerciantes e consultores/especialistas).	id_usuario (PK), nome, email, senha_hash, tipo_usuario (Comerciante, Consultor), plano (Gratuito, Premium), ra, curso, telefone
NEGOCIO	Armazena informações sobre o pequeno negócio do comerciante.	id_negocio (PK), id_usuario (FK), nome_negocio, tipo_negocio (Ex: Bar, Padaria, Mercadinho), descricao_inicial
DIAGNOSTICO	Armazena as respostas do formulário guiado e o resultado.	id_diagnostico (PK), id_negocio (FK), data_diagnostico, respostas_json (para as respostas detalhadas), pontuacao_digital
ROADMAP	Armazena o roteiro de digitalização gerado automaticamente.	id_roadmap (PK), id_diagnostico (FK), data_geracao, etapas_detalhadas (Plano de Ação), status (Em andamento, Concluído)
CONSULTORIA	Armazena dados dos agendamentos online e histórico do suporte (Chat e Vídeo).	id_consultoria (PK), id_usuario_comerciante (FK), id_usuario_consultor (FK), data_hora_agendamento, link_reuniao (para vídeo), historico_chat, observacoes_suporte
CONTEUDO_EDUCATIVO	Armazena artigos, vídeos e materiais para capacitação.	id_conteudo (PK), titulo, categoria, url_acesso, versao_acesso (Gratuito, Premium)

Legenda:

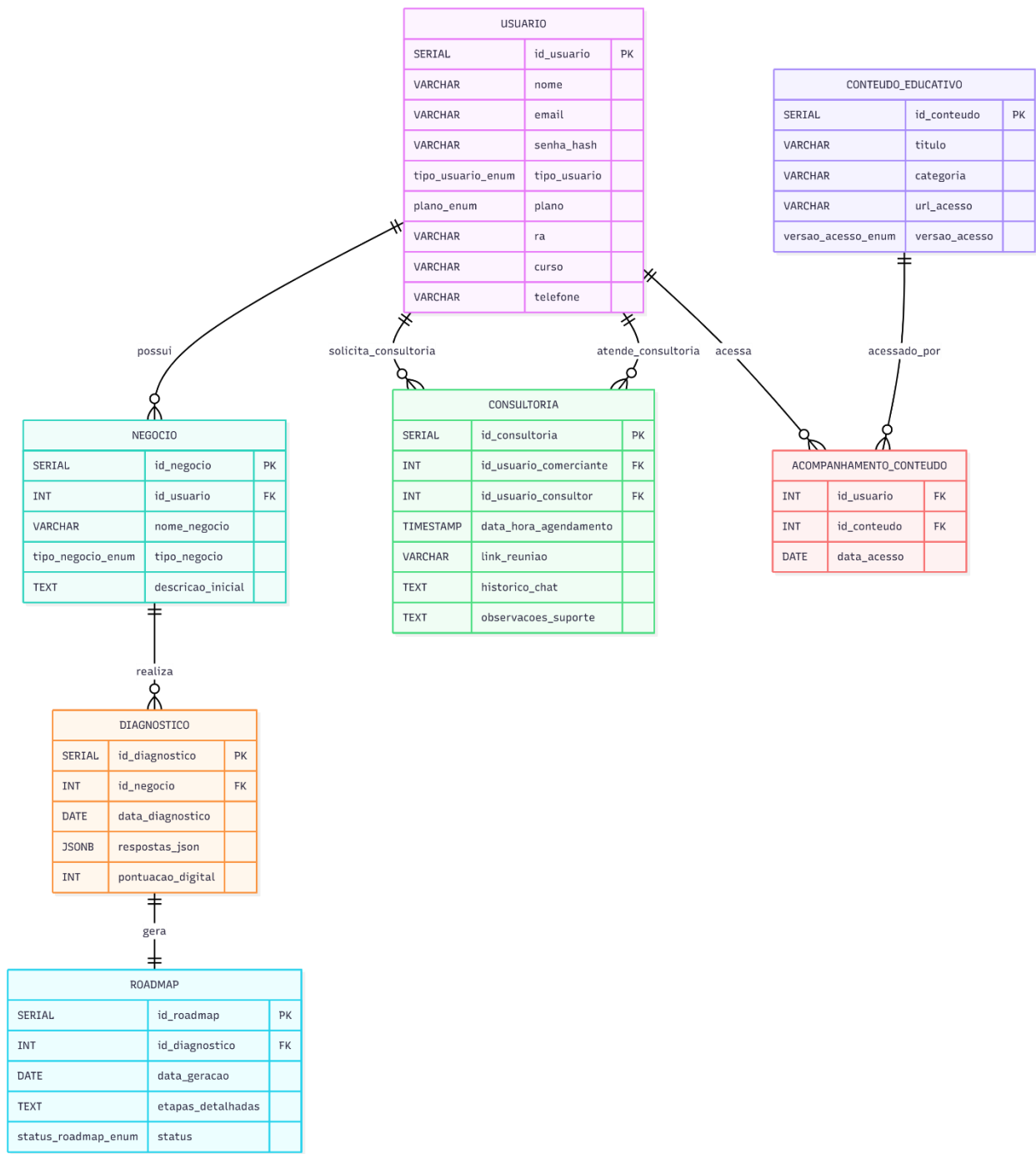
- **PK:** Chave Primária (Identificador único da tabela)
- **FK:** Chave Estrangeira (Ligação com outra tabela)
- **JSON:** Tipo de dado para armazenar estruturas de texto complexas (como as respostas do formulário ou os relatórios detalhados).

Observação sobre a Persistência em Java: Para um servidor em Java, a forma ideal de interagir com este banco de dados relacional é através de uma tecnologia ORM (Object-Relational Mapping), como **JPA (Java Persistence API)** e sua implementação mais popular, o **Hibernate**. As entidades listadas acima (USUARIO, NEGOCIO, DIAGNOSTICO, etc.) serão mapeadas diretamente para Classes Java com o mesmo nome, usando anotações JPA.

Modelo Entidade Relacional (MER) Simplificado:



Tabelas:



Código Fonte:

<https://github.com/OctavioNascimento23/SI-PI4-2025-T1-G05>

Instruções:

1. Baixe e instale o pgAdmin 4 (<https://www.pgadmin.org/download/>)
2. Crie um servidor
3. Crie um banco de dados
4. Abra a *Query Tool* (alt+shift+q)
5. Acesse o GitHub do projeto:
6. Execute o conteúdo do arquivo ***CreateTable.sql***
7. Caso der erro, utilizar o ***CreateTable_wAlterTable.sql***
8. Execute o conteúdo do arquivo ***Samples.sql***