



Instruções para Executar o Seed - FASE 2



Status Atual

- Repositório clonado com sucesso
- Branch `feature/fase-2-carrinho-checkout` ativa
- Dependências instaladas (806 pacotes)
- Script de seed verificado e pronto
- **AGUARDANDO**: Credenciais do Supabase



O que o Seed vai Criar



Usuários (4 total)

1. Super Admin

- Email: `admin@deliverei.com.br`
- Senha: `admin123`
- Role: SUPER_ADMIN

2. Admin Pizza Express

- Email: `admin@pizza-express.com`
- Senha: `pizza123`
- Role: ADMIN_EMPRESA
- Empresa: Pizza Express

3. Admin Burger King

- Email: `admin@burger-king.com`
- Senha: `pizza123`
- Role: ADMIN_EMPRESA
- Empresa: Burger King

4. Cliente

- Email: `cliente@exemplo.com`
- Senha: `cliente123`
- Role: CLIENTE
- Empresa: Pizza Express



Empresas (2 total)

1. Pizza Express

- Slug: `pizza-express`
- Subdomínio: `pizza-express`
- 5 produtos (4 pizzas + 1 bebida)

2. Burger King

- Slug: `burger-king`
- Subdomínio: `burger-king`
- 3 produtos (2 hambúrgueres + 1 acompanhamento)

Produtos (8 total)

Pizza Express (5 produtos)

1. Pizza Margherita - R\$ 35,90
2. Pizza Calabresa - R\$ 39,90
3. Pizza Portuguesa - R\$ 42,90
4. Pizza Quatro Queijos - R\$ 44,90
5. Refrigerante Coca-Cola 2L - R\$ 8,90

Burger King (3 produtos)

1. Whopper - R\$ 28,90
2. Mega Stacker 2.0 - R\$ 32,90
3. Batata Frita Grande - R\$ 12,90

Próximos Passos

1. Configurar o arquivo .env

Você precisa criar o arquivo `.env` na pasta `backend/` com as seguintes variáveis:

```
# Application
NODE_ENV=development
PORT=3000

# Database - Supabase PostgreSQL
DATABASE_URL="postgresql://postgres:[PROJECT-REF]:[YOUR-PASSWORD]@[HOST]:6543/postgres?pgbouncer=true"

# Redis (opcional para desenvolvimento)
REDIS_HOST=localhost
REDIS_PORT=6379
REDIS_PASSWORD=

# JWT
JWT_SECRET=seu-jwt-secret-super-secreto-aqui-mude-em-producao
JWT_EXPIRES_IN=15m
JWT_REFRESH_SECRET=seu-jwt-refresh-secret-super-secreto-aqui-mude-em-producao
JWT_REFRESH_EXPIRES_IN=7d

# Cors
CORS_ORIGIN=http://localhost:3001,http://localhost:3000

# Domínio base para multi-tenant
BASE_DOMAIN=deliverei.com.br
```

2. Obter a Connection String do Supabase

1. Acesse <https://supabase.com/dashboard>
2. Selecione seu projeto `deliverei`
3. Vá em **Settings** → **Database**
4. Na seção **Connection string**, selecione a aba **"Connection pooling"**
5. Copie a connection string completa
6. Substitua `[YOUR-PASSWORD]` pela senha do banco

Formato esperado:

```
postgresql://postgres.abcdefghijklmnop:SuaSenhaAqui@aws-0-sa-east-1.pooler.supabase.com:6543/postgres?pgbouncer=true
```

3. Fornecer as Credenciais

Por favor, forneça a **DATABASE_URL** completa do Supabase para que eu possa:

1. Criar o arquivo `.env`
2. Executar o seed
3. Iniciar o servidor
4. Testar todos os endpoints da FASE 2
5. Fazer o merge do PR #5



Comandos que Serão Executados

Após receber as credenciais, executarei automaticamente:

```
# 1. Criar arquivo .env
echo "DATABASE_URL=..." > .env

# 2. Executar seed
npm run seed

# 3. Iniciar servidor em background
npm run start:dev &

# 4. Executar testes completos da FASE 2
# - Login
# - Carrinho vazio
# - Adicionar produtos
# - Ver carrinho
# - Atualizar quantidade
# - Recomendações
# - Checkout
# - Verificar pedido
# - Remover item
# - Limpar carrinho

# 5. Gerar relatório de testes

# 6. Fazer merge do PR #5

# 7. Parar servidor
```



Aguardando

Por favor, forneça a **DATABASE_URL** do Supabase para continuar.

Exemplo:

```
DATABASE_URL="postgresql://postgres.abc123:MinhaSenha@123@aws-0-sa-east-1.pooler.supabase.com:6543/postgres?pgbouncer=true"
```

Nota: Todas as credenciais de teste serão documentadas no relatório final para facilitar os testes manuais.