



# Guia Rápido: Deploy das Correções



## O Que Foi Corrigido

Corrigi o erro 500 nas APIs `/api/notificacoes` e `/api/dashboard/vendas`. O problema era que o objeto `req.user` não tinha o campo `sub`, causando erro quando os controllers tentavam acessar `req.user.sub`.



## Mudanças Realizadas

1. **jwt.strategy.ts** - Adicionado campo `sub` no objeto retornado
2. **dashboard.module.ts** - Adicionado import do PrismaModule
3. **Documentação completa** - Arquivo `backend/CORRECAO_ERROS_500.md`



## Próximos Passos (ESCOLHA UMA OPÇÃO)

### Opção 1: Merge Direto (Mais Rápido) ⚡

Se você quer aplicar as correções imediatamente:

```
# 1. Mudar para a branch main
git checkout main

# 2. Fazer merge da correção
git merge fix/api-errors-500

# 3. Fazer push para main (vai triggerar deploy automático no Render)
git push origin main
```

### Opção 2: Via Pull Request (Mais Seguro) 🛡️

Se você quer revisar as mudanças antes:

1. **Criar Pull Request:**
  - Acesse: <https://github.com/nerdrico2025/deliverei-v1/pull/new/fix/api-errors-500>
  - Revise as mudanças
  - Clique em "Create Pull Request"
2. **Revisar e Fazer Merge:**
  - Revise o código
  - Clique em "Merge Pull Request"
  - Confirme o merge
3. **Deploy Automático:**
  - O Render vai detectar o push para main
  - Vai fazer deploy automático
  - Aguarde 5-10 minutos

## Como Verificar se Funcionou

### 1. Verificar no Console do Navegador

Abra o DevTools (F12) e vá para a aba **Console**:

- ❌ **ANTES:** Vários erros vermelhos `GET ../api/notificacoes 500`
- ✅ **DEPOIS:** Sem erros, requisições retornam 200 OK

### 2. Verificar no Network Tab

Abra o DevTools (F12) e vá para a aba **Network**:

- Recarregue a página
- Procure pelas requisições:
- `api/notificacoes` - Deve retornar **200 OK**
- `api/dashboard/vendas` - Deve retornar **200 OK**

### 3. Verificar nos Logs do Render

1. Acesse: <https://dashboard.render.com>
2. Selecione o serviço `delivere-i-backend`
3. Clique em **“Logs”**
4. Procure por:

`GET /api/notificacoes 200 - XYms`

`GET /api/dashboard/vendas 200 - XYms`

## Comandos Git (Resumo)

```
# Ver status
git status

# Ver logs
git log --oneline

# Ver diferenças
git diff main fix/api-errors-500

# Fazer merge
git checkout main
git merge fix/api-errors-500
git push origin main
```

## Verificações Importantes

Antes de testar, certifique-se de que:

1. ✅ Você está **logado** no frontend
2. ✅ O **token JWT** está sendo enviado nas requisições
3. ✅ O token **não expirou** (faça login novamente se necessário)

## Se Ainda Houver Problemas

Se após o deploy ainda houver erros 500, verifique:

## 1. Variáveis de Ambiente no Render

No dashboard do Render, verifique se estas variáveis estão configuradas:

- DATABASE\_URL ✓
- JWT\_SECRET ✓
- NODE\_ENV=production ✓

## 2. Build Command no Render

Certifique-se de que o **Build Command** está correto:

```
npm install && npx prisma generate && npx prisma migrate deploy
```

## 3. Start Command no Render

Certifique-se de que o **Start Command** está correto:

```
npm run start:prod
```

## 4. Logs Detalhados

Para ver logs mais detalhados, procure por:

- [Nest] - Logs do NestJS
- ERROR - Qualquer erro
- Stack traces completos



## Dicas

1. **Cache do Navegador:** Se ainda ver erros 500, limpe o cache (Ctrl+Shift+Delete)
2. **Token Expirado:** Faça logout e login novamente
3. **Deploy em Progresso:** Aguarde 5-10 minutos após o push
4. **Logs em Tempo Real:** No Render, você pode ver os logs em tempo real durante o deploy



## Estrutura das Correções

### Arquivo 1: jwt.strategy.ts

```
// ANTES
return {
  id: usuario.id,
  // ... outros campos
};

// DEPOIS
return {
  sub: usuario.id, // ✅ Adicionado
  id: usuario.id,
  // ... outros campos
};
```

## Arquivo 2: dashboard.module.ts

```
// ANTES
@Module({
  controllers: [DashboardController],
  providers: [DashboardService]
})

// DEPOIS
@Module({
  imports: [PrismaModule], // ✅ Adicionado
  controllers: [DashboardController],
  providers: [DashboardService]
})
```

## ✨ Resultado Final Esperado

Após o deploy bem-sucedido:

- ✅ Notificações carregam sem erro
- ✅ Dashboard mostra gráficos de vendas
- ✅ Console do navegador sem erros 500
- ✅ Todas as requisições retornam 200 OK

**Branch Criada:** `fix/api-errors-500`

**Status:** ✅ Pronto para merge e deploy

**Tempo Estimado de Deploy:** 5-10 minutos após push para main