





# Correção Definitiva do Prisma Client no Vercel

## Problema Identificado

O erro persistente no Vercel era causado por **cache do Prisma Client**:

```
Type error: 'lgpdConsentAt' does not exist in type 'UserSelect<DefaultArgs>'
```

## Causa Raiz

1.  O campo `lgpdConsentAt` existe no `schema.prisma`
2.  A migration `20251031191431_add_lgpd_consent` foi aplicada
3.  O Vercel estava usando cache do `node_modules/.prisma` antigo
4.  O TypeScript verificava contra um Prisma Client desatualizado

## Solução Implementada

### 1 Arquivo: `prisma/schema.prisma`

Removido output customizado:

```
generator client {
  provider = "prisma-client-js"
  binaryTargets = ["native", "linux-musl-arm64-openssl-3.0.x"]
  // output removido - usa padrão do Prisma
}
```

### 2 Arquivo: `scripts/postinstall.sh` (NOVO)

Script que limpa cache antes de gerar:



```
#!/bin/bash

echo "
```

```
{
  "buildCommand": "npm run build",
  "installCommand": "npm install --legacy-peer-deps && bash scripts/postinstall.sh"
}
```

## Como Funciona





### Fluxo de Build no Vercel:

1. Clone do repositório  

2. `npm install --legacy-peer-deps` (instala dependências)  

3. `bash scripts/postinstall.sh` (LIMPA + REGENERA Prisma)  







4. `npm run build` (build do **Next.js**)  

5. Deploy concluído 

### Por que isso resolve:

-  **Limpeza forçada:** Remove todo cache anterior do Prisma
-  **Regeneração limpa:** Gera Prisma Client com schema atualizado
-  **Sem cache corrompido:** TypeScript vê campos corretos
-  **Build consistente:** Funciona tanto local quanto no Vercel

## Validações Realizadas

-  Build **local** passou
-  TypeScript check passou
-  Prisma Client regenerado corretamente
-  Script `postinstall.sh` testado
-  Commit criado
-  Push para GitHub concluído






## Commits Aplicados

78035d5 - fix: Resolve Prisma Client cache no Vercel **definitivamente**  
 5083fc2 - Corrige Prisma Client path Vercel  
 c733c0e - fix: Corrige caminho do Prisma Client para compatibilidade com Vercel






## Próximo Deploy do Vercel

---

### O que esperar:

1.  **Instalação:** npm install executado normalmente
2.  **Limpeza:** Cache do Prisma removido
3.  **Regeneração:** Prisma Client gerado do zero
4.  **Build:** TypeScript reconhece IgpdConsentAt
5.  **Deploy:** Sucesso!




### Logs esperados:

```
 Limpando cache do Prisma Client...  
 Gerando Prisma Client...  
 Generated Prisma Client (v6.7.0) to ./node_modules/@prisma/client  
 Prisma Client gerado com sucesso
```




---

## Prevenção de Problemas Futuros

### Evitado:

-  Caminhos absolutos no `output` do Prisma
-  Cache inconsistente entre ambientes
-  TypeScript desatualizado com schema

### Garantido:

-  Regeneração limpa em cada build
-  Compatibilidade local e Vercel
-  Campos do schema sempre reconhecidos

---

## Comandos para Replicar Localmente

```
# 1. Clone e instale  
git clone https://github.com/lebervinicius-dev/orcamento-planejado.git  
cd orcamento-planejado/nextjs_space  
npm install --legacy-peer-deps  
  
# 2. Execute postinstall manualmente  
bash scripts/postinstall.sh  
  
# 3. Build  
npm run build
```

## Status Final

Item	Status
✓ Problema identificado	Concluído
✓ Solução implementada	Concluído
✓ Código testado	Concluído
✓ Push para GitHub	Concluído
🕒 Deploy Vercel	Em andamento

## Próximos Passos

1. **Aguarde 2-5 minutos** - Vercel está fazendo build
2. **Monitore os logs** no dashboard do Vercel
3. **Verifique sucesso** - Build deve passar agora
4. **Teste a aplicação** após deploy

**Data:** 31/10/2025

**Commit:** 78035d5

**Status:** ✓ Correção definitiva aplicada