

# Análise Detalhada: Fase 5 - Fuso Horário e Regionalização

## Resumo Executivo

Status da Fase 5: NÃO IMPLEMENTADA (0% concluído)

A Fase 5, conforme definida no plano de implementação completo, tem como objetivo fornecer suporte multi-regional completo ao OrganiZen, permitindo que cada empresa configure: - Fuso horário (timezone) - Formatos de data e hora - Primeiro dia da semana - Moeda padrão

## O Que Foi Definido no Plano de Implementação

### 5.1 Database Schema

```
model Company {  
    // ... campos existentes  
    timezone          String  @default("America/Sao_Paulo")  
    dateFormat        String  @default("DD/MM/YYYY")  
    timeFormat        String  @default("24h") // "12h" ou "24h"  
    firstDayOfWeek   Int    @default(0) // 0 = Domingo, 1 = Segunda  
    currency          String  @default("BRL")  
}
```

### 5.2 API Endpoints Necessários

- GET /api/settings/regional - Buscar configurações
- PUT /api/settings/regional - Atualizar configurações

### 5.3 UI Components Necessários

- Seletor de fuso horário (timezone picker)
- Seletor de formato de data
- Seletor de formato de hora (12h/24h)
- Seletor de primeiro dia da semana
- Seletor de moeda
- Preview de formatação

### 5.4 Funcionalidades Esperadas

- Configurar fuso horário da empresa
- Definir formato de data (DD/MM/YYYY, MM/DD/YYYY, etc.)
- Definir formato de hora (12h/24h)
- Definir primeiro dia da semana
- Definir moeda padrão
- Preview em tempo real das mudanças

### 5.5 Lógica de Aplicação Esperada

- Converter todas as datas para timezone configurado
- Formatar datas conforme formato selecionado
- Aplicar formato de hora globalmente
- Ajustar calendários para primeiro dia configurado

## O Que JÁ Foi Implementado (Parcial)

### 1. Database Schema (Parcial - ~20%)

No arquivo `prisma/schema.prisma`, o model `Company` possui apenas:

```
model Company {
    id          String      @id @default(cuid())
    name        String
    email       String      @unique

    // Configurações Regionais (Fase 1) - PARCIAL
    defaultLanguage String      @default("pt") // pt, en, es, fr

    // FALTAM:
    // timezone           String      @default("America/Sao_Paulo")
    // dateFormat         String      @default("DD/MM/YYYY")
    // timeFormat         String      @default("24h")
    // firstDayOfWeek    Int        @default(0)
    // currency           String      @default("BRL")

    // ... outros campos
}
```

**Status:** Apenas o campo `defaultLanguage` foi implementado na Fase 1. Os demais 5 campos da Fase 5 estão **ausentes**.

### 2. API Endpoints (0%)

**Não implementados:** - `/api/settings/regional` (GET e PUT)

A estrutura atual de APIs em `/app/api/settings/` contém: - `/api/settings/company` (para info gerais)  
- `/api/settings/security` - `/api/settings/notifications` - `/api/settings/task-statuses` -  
`/api/settings/task-priorities` - `/api/settings/task-tags`

Mas **não há** endpoint dedicado para configurações regionais.

### 3. UI Components (0%)

**Não implementados:**

A estrutura atual de páginas em `/app/settings/` contém: - `/settings/branding` - `/settings/company` -  
`/settings/email-templates` - `/settings/security` - `/settings/notifications` - `/settings/tasks`  
- `/settings/sessions` - `/settings/testimonials`

Mas **não há**: - `/settings/regional` ou - `/settings/regional-preferences` ou equivalente

### 4. Referências na Interface Principal

No arquivo `/app/settings/page.tsx`, existe a seção “Preferências Pessoais” que menciona:

```
{
  id: 'personal',
  title: 'Preferências Pessoais',
  description: 'Idioma, fuso horário e notificações pessoais',
  icon: User,
  href: '/settings/personal',
  available: false,
```

```
        badge: 'Em breve',
    }
```

**Nota Importante:** Esta secção refere-se a **preferências PESSOAIS por utilizador** (Fase 6C), e não às configurações **GLOBAIS da empresa** (Fase 5).

A Fase 5 deveria ter uma entrada separada para configurações regionais **da empresa**, algo como:

```
{
  id: 'regional',
  title: 'Configurações Regionais',
  description: 'Fuso horário, formato de data/hora e moeda',
  icon: Globe, // ou similar
  href: '/settings/regional',
  available: true,
  badge: null,
}
```

Mas esta entrada **não existe**.

---

## O Que NÃO Foi Implementado (Fase 5 Completa)

### 1. Schema do Banco de Dados

**Faltam 5 campos no model Company:** - `timezone: String` - Fuso horário da empresa - `dateFormat: String` - Formato de data (DD/MM/YYYY, MM/DD/YYYY, etc.) - `timeFormat: String` - Formato de hora (12h ou 24h) - `firstDayOfWeek: Int` - Primeiro dia da semana (0=Domingo, 1=Segunda) - `currency: String` - Moeda padrão (BRL, USD, EUR, etc.)

### 2. API Endpoints

**Faltam:** - GET `/api/settings/regional` - Buscar configurações regionais da empresa - PUT `/api/settings/regional` - Atualizar configurações regionais

**Funcionalidades esperadas nos endpoints:** - Validar timezone contra lista de timezones válidos (IANA) - Validar formato de data contra padrões suportados - Validar formato de hora (12h/24h) - Validar primeiro dia da semana (0-6) - Validar código de moeda (ISO 4217)

### 3. UI Components

**Faltam todos os componentes de UI:**

**3.1 Página Principal:** `/app/settings/regional/page.tsx` Deveria conter: - Header com título e descrição - Card para “Fuso Horário” - Card para “Formato de Data e Hora” - Card para “Calendário” - Card para “Moeda” - Preview em tempo real das mudanças

#### 3.2 Componente: Timezone Picker

```
// /components/settings/TimezonePicker.tsx
// - Busca de timezones por região
// - Listagem de timezones comuns
// - Exibição de offset UTC
// - Preview de hora atual no timezone selecionado
```

### **3.3 Componente: Date Format Selector**

```
// /components/settings/DateFormatSelector.tsx  
// - Opções: DD/MM/YYYY, MM/DD/YYYY, YYYY-MM-DD, etc.  
// - Preview com data exemplo  
// - Suporte a separadores customizados
```

### **3.4 Componente: Time Format Selector**

```
// /components/settings/TimeFormatSelector.tsx  
// - Toggle entre 12h e 24h  
// - Preview de hora exemplo
```

### **3.5 Componente: First Day of Week Selector**

```
// /components/settings/FirstDaySelector.tsx  
// - Radio buttons para Domingo/Segunda  
// - Preview de calendário mini
```

### **3.6 Componente: Currency Selector**

```
// /components/settings/CurrencySelector.tsx  
// - Busca de moedas por código ou nome  
// - Listagem de moedas comuns (BRL, USD, EUR, etc.)  
// - Exibição de símbolo da moeda  
// - Preview de valores formatados
```

## **4. Lógica de Aplicação**

Faltam as integrações:

### **4.1 Conversão de Timezones**

- Biblioteca sugerida: `date-fns-tz` ou `luxon`
- Converter todas as datas do banco (UTC) para timezone da empresa
- Exibir datas conforme timezone configurado
- Salvar sempre em UTC no banco

### **4.2 Formatação de Datas**

- Aplicar formato configurado em:
  - Listagens de tarefas
  - Calendário
  - Relatórios
  - Mensagens
  - Logs de auditoria
  - Exportações

### **4.3 Formatação de Horas**

- Aplicar formato 12h/24h em:
  - Relógio do sistema
  - Horários de turnos
  - Horários de eventos
  - Timestamps

#### 4.4 Primeiro Dia da Semana

- Ajustar componentes de calendário:
  - Calendário principal
  - Date pickers
  - Visualizações semanais

#### 4.5 Formatação de Moeda

- Aplicar moeda configurada em:
  - Valores de faturamento (se implementado)
  - Relatórios financeiros
  - Exportações
  - Formatação conforme locale

### 5. Testes

**Não implementados:** - Configurar timezone e validar conversão de datas - Testar diferentes formatos de data/hora - Validar primeiro dia da semana no calendário - Verificar formatação de valores monetários

### 6. Checkpoint

**Não criado:** - Checkpoint: "Regional and timezone settings"

---

## Impacto da Não Implementação

### Problemas Atuais

1. **Timezones inconsistentes:** Todas as datas são exibidas em UTC ou timezone do servidor, não da empresa
2. **Formato fixo:** Datas e horas seguem um único formato, sem adaptação regional
3. **Calendários não configuráveis:** Primeiro dia da semana é fixo
4. **Sem suporte a moeda:** Valores não são formatados conforme região

### Áreas Afetadas

- Calendário (eventos, turnos, feriados)
  - Tarefas (prazos, datas de criação)
  - Mensagens (timestamps)
  - Relatórios (datas e valores)
  - Logs de auditoria
  - Notificações (horários de envio)
  - Exportações (formato de dados)
- 

## Checklist de Implementação (0% Completo)

### Database (0/1)

- Adicionar 5 campos ao model Company:
  - `timezone: String`
  - `dateFormat: String`
  - `timeFormat: String`
  - `firstDayOfWeek: Int`
  - `currency: String`

- Criar e executar migration
- Atualizar seed.ts com valores padrão

### API Endpoints (0/2)

- Criar /api/settings/regional/route.ts
  - Implementar GET (buscar configurações)
  - Implementar PUT (atualizar configurações)
  - Validar permissões (apenas ADMIN/OWNER)
  - Validar dados de entrada

### UI Components (0/7)

- Criar /app/settings/regional/page.tsx
- Criar /components/settings/TimezonePicker.tsx
- Criar /components/settings/DateFormatSelector.tsx
- Criar /components/settings/TimeFormatSelector.tsx
- Criar /components/settings/FirstDaySelector.tsx
- Criar /components/settings/CurrencySelector.tsx
- Criar /components/settings/RegionalPreview.tsx

### Integração (0/5)

- Instalar dependências (date-fns-tz ou luxon)
- Criar helpers de formatação:
  - formatDate() - Formata data conforme config
  - formatTime() - Formata hora conforme config
  - formatCurrency() - Formata moeda conforme config
- Atualizar componentes existentes:
  - Calendário principal
  - Date pickers
  - Listagem de tarefas
  - Timestamps de mensagens
- Atualizar lógica de backend:
  - Conversão de timezone em queries
  - Formatação de datas em responses

### Interface Principal (0/1)

- Adicionar entrada para “Configurações Regionais” em /app/settings/page.tsx

### Testes (0/4)

- Testar configuração de timezone
- Testar formatos de data/hora
- Testar primeiro dia da semana no calendário
- Testar formatação de moeda

### Documentação (0/1)

- Criar checkpoint: "Regional and timezone settings"
-

## Recomendações de Implementação

### Ordem Sugerida

1. **Schema do Banco** (30 min)
  - Adicionar campos ao model Company
  - Criar e executar migration
  - Atualizar seed.ts
2. **API Endpoints** (1h)
  - Criar /api/settings/regional
  - Implementar GET e PUT
  - Adicionar validações
3. **UI Components Básicos** (2h)
  - Criar página /settings/regional
  - Implementar selectors simples
  - Adicionar preview
4. **Integração Inicial** (1-2h)
  - Instalar dependências
  - Criar helpers de formatação
  - Atualizar 2-3 componentes críticos (calendário, tarefas)
5. **Integração Completa** (2-3h)
  - Atualizar todos os componentes restantes
  - Aplicar formatação em toda a aplicação
6. **Testes e Checkpoint** (30 min)
  - Testar funcionalidades
  - Criar checkpoint

**Tempo Total Estimado:** 2-3 horas (conforme plano original)

### Prioridade

#### MÉDIA-ALTA

Esta fase é importante para:  
- Empresas internacionais ou multi-regionais  
- Empresas que precisam de conformidade com formatos locais  
- Melhorar UX para usuários em diferentes fusos horários

Mas não é bloqueante para funcionalidades core.

### Riscos

- **Baixo Risco:** Não afeta funcionalidades existentes
  - **Impacto Médio:** Melhora significativa de UX após implementação
  - **Reversível:** Pode ser revertido facilmente via checkpoint
- 

## Conclusão

**Status Atual:** Fase 5 está **0% implementada**

**O que existe:** Apenas o campo defaultLanguage da Fase 1

**O que falta:** Todos os 5 campos principais, APIs, UI e lógica de aplicação

**Próximos Passos:** 1. Decidir se implementar Fase 5 agora ou continuar com outras fases 2. Se implementar, seguir ordem sugerida acima 3. Criar checkpoint após conclusão

**Tempo estimado para implementação completa:** 2-3 horas

---

**Documento gerado em:** 27 de Outubro de 2025 **Projeto:** OrganiZen **Versão:** 1.0