

Implementación Completa: Sistema de Plantillas de Mensajes

Fecha: 31 de octubre de 2025

Estado: ☒ Completado y en GitHub

Última revisión: 31-OCT-2025 01:50 UTC

Resumen Ejecutivo

Se implementó un sistema completo de gestión de plantillas de mensajes para múltiples canales de comunicación (SMS LabMobile, WhatsApp, Chatwoot) y se corrigió un bug crítico que impedía visualizar los módulos de Chatwoot y Plantillas de Mensajes en el menú.



Módulos Implementados

1. Sistema de Plantillas de Mensajes

Funcionalidades:

- ☒ Gestión CRUD completa de plantillas
- ☒ Soporte multi-canal (SMS, WhatsApp, Chatwoot, Email, In-App)
- ☒ Sistema de variables dinámicas personalizables
- ☒ Límite de caracteres específico para SMS (160 caracteres)
- ☒ Categorización por tipo de mensaje
- ☒ Templates predefinidos con seed automático
- ☒ API RESTful completa
- ☒ Interfaz administrativa moderna

Canales soportados:

- **SMS (LabMobile):** Máximo 160 caracteres
- **WhatsApp:** Sin límite de caracteres
- **Chatwoot:** Para chat en tiempo real
- **Email:** Para comunicaciones formales
- **In-App:** Notificaciones dentro de la aplicación

Variables dinámicas disponibles:

```
{cliente_nombre}  
{cliente_apellido}  
{prestamo_monto}  
{prestamo_plazo}  
{pago_fecha}  
{pago_monto}  
{empresa_nombre}  
{asesor_nombre}
```

Archivos Implementados

Base de Datos

Archivo: app/prisma/schema.prisma

```
model MessageTemplate {
  id          String   @id @default(cuid())
  name        String   // Nombre descriptivo de la plantilla
  channel     MessageChannel // SMS, WHATSAPP, CHATWOOT, EMAIL, INAPP
  category    String   // PAYMENT_REMINDER, APPROVAL_NOTICE, etc.
  subject     String?  // Para email
  body        String   // Contenido del mensaje con variables {variable}
  variables   String[] // Lista de variables disponibles
  maxLength   Int?    // Límite de caracteres (160 para SMS)
  isActive    Boolean  @default(true)
  metadata    Json?    // Información adicional
  createdAt   DateTime @default(now())
  updatedAt   DateTime @updatedAt

  @@map("message_templates")
}

enum MessageChannel {
  SMS      // LabMobile SMS
  WHATSAPP // WhatsApp Business
  CHATWOOT // Chat en tiempo real
  EMAIL    // Correo electrónico
  INAPP    // Notificaciones in-app
}
```

API Endpoints

Archivo: app/api/admin/message-templates/route.ts

```
GET    /api/admin/message-templates // Listar todas las plantillas
POST   /api/admin/message-templates // Crear nueva plantilla
GET    /api/admin/message-templates/:id // Obtener plantilla específica
PATCH /api/admin/message-templates/:id // Actualizar plantilla
DELETE /api/admin/message-templates/:id // Eliminar plantilla
```

Respuesta de ejemplo:

```
{
  "id": "clxxx...",
  "name": "Recordatorio de Pago",
  "channel": "SMS",
  "category": "PAYMENT_REMINDER",
  "body": "Hola {cliente_nombre}, te recordamos que tienes un pago pendiente de $
{pago_monto} con vencimiento el {pago_fecha}.",
  "variables": ["cliente_nombre", "pago_monto", "pago_fecha"],
  "maxLength": 160,
  "isActive": true
}
```

Frontend

Archivo: app/components/admin/message-templates-management.tsx

Componente React completo con:

- Lista de plantillas con filtros por canal y categoría
- Formulario de creación/edición con validación
- Contador de caracteres en tiempo real para SMS
- Preview de mensaje con variables
- Sistema de confirmación para eliminación
- Indicador visual de canal con colores distintivos
- Soporte completo de dark mode

Página: `app/app/admin/message-templates/page.tsx`

Ruta: `/admin/message-templates`

Scripts de Seed

Archivo: `app/scripts/seed-message-templates.ts`

Plantillas predefinidas incluidas:

1. Recordatorio de pago (SMS)
2. Confirmación de aprobación (WhatsApp)
3. Mensaje de bienvenida (Chatwoot)
4. Notificación de vencimiento (Email)
5. Alerta de mora (In-App)



Bug Crítico Corregido

Problema

Los módulos de **Chatwoot** y **Plantillas de Mensajes** no aparecían en el menú a pesar de:

- Estar correctamente implementados
- Tener rutas funcionales
- Estar configurados en los componentes de navegación

Causa Raíz

El módulo `notifications_templates` no existía en el archivo de seed de módulos (`app/scripts/seed-modules.js`).

Sin este módulo en la base de datos, el sistema de control de módulos PWA (`ModuleWrapper`) ocultaba automáticamente los enlaces del menú.

Solución

Se agregó el módulo faltante al seed:

```
{
  moduleKey: 'notifications_templates',
  name: 'Plantillas de Mensajes',
  description: 'Gestión de plantillas para SMS, WhatsApp, Chatwoot y otros canales',
  category: 'NOTIFICATIONS',
  status: 'ENABLED',
  isCore: false,
  requiredFor: [],
  availableFor: ['ADMIN', 'ASESOR'],
  icon: 'Mail',
  route: '/admin/message-templates',
  sortOrder: 52,
}
```

Commits Realizados

5056e1e - fix(prisma): Corregir ruta absoluta a relativa en schema.prisma output
 3c39418 - fix: Convertir yarn.lock a archivo regular (detectado por pre-push check)
 7b765cc - fix(modules): Agregar módulo faltante notifications_templates
 448e017 - Sistema plantillas mensajes SMS/WhatsApp/Chatwoot

Estado en GitHub:  Todos los commits pusheados exitosamente

Instrucciones de Deployment

1. Preparación en EasyPanel

Antes de hacer el rebuild, verifica que tienes configuradas las variables de entorno necesarias:

```
# Base de datos (requerido)
DATABASE_URL=postgresql://user:password@host:5432/database

# Variables de la aplicación (requerido)
NEXTAUTH_SECRET=tu_secret_seguro
NEXTAUTH_URL=https://tu-dominio.com

# Para SMS LabMobile (opcional, solo si usarás SMS)
LABMOBILE_API_KEY=tu_api_key
LABMOBILE_SENDER_NAME=EscalaFin

# Para WhatsApp (opcional)
EVOLUTION_API_URL=https://tu-evolution-api.com
EVOLUTION_API_KEY=tu_api_key

# Para Chatwoot (opcional)
CHATWOOT_BASE_URL=https://tu-chatwoot.com
CHATWOOT_WEBSITE_TOKEN=tu_token
```

2. Rebuild en EasyPanel

1. Pull del último commit:

- Ir a tu aplicación en EasyPanel

- Navegar a la sección “Source”
- Click en “Pull” para obtener los últimos cambios
- Verificar que aparezca el commit `5056e1e`

2. Clear Build Cache (recomendado):

- Click en “Advanced”
- Seleccionar “Clear build cache”
- Esto garantiza un build limpio

3. Rebuild:

- Click en “Rebuild”
- Esperar a que el build complete (5-10 minutos aprox.)

4. Monitorear Logs:

```
bash
# Deberías ver durante el startup:
🌱 Seeding PWA modules...
Processing module: Plantillas de Mensajes (notifications_templates)
✅ Permissions configured for 2 roles
🎉 PWA modules seeded successfully!
```

3. Verificación Post-Deployment

1. Iniciar sesión como ADMIN o ASESOR

2. Verificar menú de Comunicación:

Deberías ver la estructura completa:

```
Comunicación
├── WhatsApp
│   ├── Mensajes
│   └── Recargas
├── Chat
│   └── Chatwoot ← DEBE APARECER
└── Notificaciones
    ├── Centro de Notificaciones
    └── Plantillas de Mensajes ← DEBE APARECER
```

1. Acceder a Plantillas:

- Click en “Plantillas de Mensajes”
- Deberías ver la interfaz de gestión
- Verificar que aparezcan las plantillas predefinidas

2. Crear una plantilla de prueba:

- Click en “Nueva Plantilla”
- Seleccionar canal: SMS
- Nombre: “Prueba SMS”
- Cuerpo: “Hola {cliente_nombre}, este es un mensaje de prueba.”
- Verificar contador: máximo 160 caracteres
- Guardar

3. Acceder a Chatwoot:

- Click en “Chatwoot”

- Deberías ver la página de configuración
- Configurar según tus credenciales de Chatwoot

Guía de Uso

Crear una Plantilla SMS

1. Ir a `/admin/message-templates`
2. Click en **"Nueva Plantilla"**
3. Configurar:
 - Nombre: Descriptivo (ej. **"Recordatorio Pago Semanal"**)
 - Canal: SMS
 - Categoría: PAYMENT_REMINDER
 - Cuerpo: Tu mensaje con variables
4. Respetar límite de 160 caracteres
5. Guardar

Ejemplo de plantilla SMS:

Hola {cliente_nombre}, te recordamos tu pago de \${pago_monto} vence el {pago_fecha}.
EscalaFin

Longitud: 95 caracteres (dentro del límite)

Crear una Plantilla WhatsApp

1. Ir a `/admin/message-templates`
2. Click en **"Nueva Plantilla"**
3. Configurar:
 - Nombre: Descriptivo
 - Canal: WhatsApp
 - Categoría: Según corresponda
 - Cuerpo: Mensaje más extenso (**sin** límite)
4. Guardar

Ejemplo de plantilla WhatsApp:

¡Felicidades {cliente_nombre}! 🎉

Tu solicitud de préstamo por \${prestamo_monto} ha sido APROBADA.

Detalles:

- Monto: \${prestamo_monto}
- Plazo: {prestamo_plazo} meses
- Tu asesor: {asesor_nombre}

Nos pondremos en contacto contigo pronto para finalizar el proceso.

Gracias por confiar en EscalaFin ❤️

Usar Variables en Plantillas

Las variables se reemplazan automáticamente al enviar el mensaje:






```
// En tu código de notificaciones:
const template = await getTemplate('PAYMENT_REMINDER', 'SMS');
const message = replaceVariables(template.body, {
  cliente_nombre: 'Juan',
  pago_monto: '500',
  pago_fecha: '15-Nov-2025'
});

// Resultado:
// "Hola Juan, te recordamos tu pago de $500 vence el 15-Nov-2025. EscalaFin"
```

Interfaz Visual




Indicadores de Canal

Cada canal tiene un color distintivo:

-  **WhatsApp**: Verde
-  **SMS**: Azul
-  **Chatwoot**: Púrpura
-  **Email**: Rojo
-  **In-App**: Amarillo















Contador de Caracteres (SMS)

Para plantillas SMS, un contador en tiempo real muestra:

-  Verde: < 140 caracteres
-  Amarillo: 140-160 caracteres
-  Rojo: > 160 caracteres (no permitido guardar)

Control de Acceso

Permisos por Rol

Acción	ADMIN	ASESOR	CLIENTE
Ver plantillas			
Crear plantillas			
Editar plantillas			
Eliminar plantillas			
Usar plantillas			N/A

Mantenimiento y Soporte

Logs a Monitorear

```
# Durante el startup (seed de módulos)
🌱 Seeding PWA modules...
✅ Module: Plantillas de Mensajes created/updated

# Durante creación de plantilla
POST /api/admin/message-templates
✅ Template created: [ID]

# Durante uso de plantilla
🔗 Sending message using template: [NAME]
✅ Message sent successfully
```

Troubleshooting Común

Problema: No aparece el menú de Plantillas de Mensajes

Solución:

1. Verificar que el módulo esté habilitado:

```
sql
```

```
SELECT * FROM pwa_modules WHERE module_key = 'notifications_templates';
```

2. Si no existe, ejecutar seed:

```
bash
```

```
node scripts/seed-modules.js
```

Problema: Error al guardar plantilla SMS > 160 caracteres

Solución:

- La validación en frontend y backend previene esto
- Reducir el texto o usar otro canal

Problema: Variables no se reemplazan

Solución:

- Verificar que uses la sintaxis correcta: {variable_nombre}
- Verificar que pases el objeto con las variables correctas

Recursos Adicionales

Documentación Relacionada

- `FIX_API_CLIENTS_CREATION_31_OCT_2025.md` - Fix del módulo faltante
- `IMPLEMENTACION_CHATWOOT_CONFIGURABLE.md` - Configuración de Chatwoot
- `RESUMEN_FIXES_PRE_DEPLOY_30_OCT_2025.md` - Fixes pre-deployment

APIs Externas

- **LabMobile SMS API:** [Documentación](https://labmobile.com/api-docs) (https://labmobile.com/api-docs)
- **WhatsApp Business API:** [Documentación](https://developers.facebook.com/docs/whatsapp) (https://developers.facebook.com/docs/whatsapp)
- **Chatwoot API:** [Documentación](https://www.chatwoot.com/docs/product/channels/api/client-apis) (https://www.chatwoot.com/docs/product/channels/api/client-apis)

✓ Checklist de Verificación

Antes de considerar la implementación completa, verifica:

- ☐ Rebuild exitoso en EasyPanel
- ☐ Módulos PWA seeded correctamente
- ☐ Menú de Comunicación muestra todos los enlaces
- ☐ Acceso a `/admin/message-templates` funcional
- ☐ Plantillas predefinidas cargadas
- ☐ Creación de nueva plantilla funciona
- ☐ Edición de plantilla funciona
- ☐ Eliminación de plantilla funciona (solo ADMIN)
- ☐ Validación de 160 caracteres funciona (SMS)
- ☐ Variables se muestran correctamente
- ☐ Acceso a `/admin/chatwoot` funcional
- ☐ Permisos por rol funcionan correctamente

Estado del Proyecto

Componente	Estado	Notas
Base de Datos	✓ Completo	Modelo MessageTemplate agregado
API Backend	✓ Completo	CRUD completo implementado
Frontend Admin	✓ Completo	UI moderna y responsive
Seed Scripts	✓ Completo	Plantillas y módulos predefinidos
Documentación	✓ Completo	Este documento
Testing	⌚ Pendiente	Pendiente de deployment
Deployment	⌚ Pendiente	Listo para rebuild en EasyPanel

Próximos Pasos

1. Inmediato:

- Hacer rebuild en EasyPanel

- Verificar que todo funcione correctamente
- Probar creación de plantillas

2. **Corto Plazo:**

- Integrar plantillas con sistema de notificaciones existente
- Implementar envío automático usando plantillas
- Agregar historial de mensajes enviados

3. **Mediano Plazo:**

- Analytics de uso de plantillas
- A/B testing de mensajes
- Personalización avanzada

Soporte

Para cualquier problema o duda:

1. **Revisar logs** en EasyPanel
 2. **Consultar documentación** de este archivo
 3. **Verificar variables de entorno**
 4. **Contactar soporte técnico** si el problema persiste
-

Documento generado: 31 de octubre de 2025

Versión: 1.0

Autor: Sistema EscalaFin

Estado: Producción Ready 