



LA RESERVA - CONSTANTES Y CONFIGURACIÓN

PARTE 6: WhatsApp y Configuraciones Finales

Versión: 1.0

Fecha: Octubre 2025

ARCHIVO: whatsapp.ts

de: `src/utils/whatsapp.ts`

```
// src/utils/whatsapp.ts

/**
 * LA RESERVA - HELPERS DE WHATSAPP
 *
 * Funciones para generar URLs y mensajes de WhatsApp
 */

import { CONTACT_INFO } from './constants';

// =====
// 1. GENERACIÓN DE URLs
// =====

/**
 * Genera URL de WhatsApp con mensaje predefinido
 *
 * @param message - Mensaje a enviar
 * @param phone - Número (opcional, usa el de contacto por defecto)
 * @returns URL de WhatsApp
 */
export function getWhatsAppUrl(message: string, phone?: string): string {
  const phoneNumber = phone || CONTACT_INFO.whatsapp.replace(/\D/g, '');
}
```

```

    const encodedMessage = encodeURIComponent(message);
    return `https://wa.me/${phoneNumber}?text=${encodedMessage}`;
}

/***
 * Abre WhatsApp en nueva ventana/tab
 *
 * @param message - Mensaje a enviar
 * @param phone - Número opcional
 */
export function openWhatsApp(message: string, phone?: string): void {
  const url = getWhatsAppUrl(message, phone);
  window.open(url, '_blank', 'noopener,noreferrer');
}

```

```

// =====
// 2. MENSAJES PREDEFINIDOS
// =====

```

```

/***
 * Genera mensaje de WhatsApp para cotización
 *
 * @param data - Datos del formulario de cotización
 * @returns Mensaje formateado
 */
export function getQuoteWhatsAppMessage(data: {
  name: string;
  eventType: string;
  eventDate: string;
  guestCount: number;
}): string {
  return `Hola! Me gustaría solicitar una cotización:

```

-  Nombre: \${data.name}
-  Tipo de evento: \${data.eventType}
-  Fecha: \${data.eventDate}
-  Cantidad de invitados: \${data.guestCount}

```

¿Podrían enviarme más información?`;
}

```

```

/***
 * Genera mensaje de WhatsApp para contacto general
 */

```

```

* @param name - Nombre opcional del cliente
* @returns Mensaje formateado
*/
export function getContactWhatsAppMessage(name?: string): string {
  const greeting = name ? `Hola! Soy ${name}.` : 'Hola!';
  return `${greeting} Me gustaría obtener más información sobre sus servicios`;
}

/**
 * Genera mensaje de WhatsApp para consulta de disponibilidad
 *
 * @param date - Fecha del evento
 * @returns Mensaje formateado
*/
export function getAvailabilityWhatsAppMessage(date: string): string {
  return `Hola! Me gustaría consultar disponibilidad para el ${date}.`;
}

/**
 * Genera mensaje de WhatsApp para consulta de servicio específico
 *
 * @param serviceName - Nombre del servicio
 * @returns Mensaje formateado
*/
export function getServiceWhatsAppMessage(serviceName: string): string {
  return `Hola! Me interesa el servicio de "${serviceName}". ¿Podrían darme más información?`;
}

/**
 * Genera mensaje de WhatsApp para consulta de paquete específico
 *
 * @param packageName - Nombre del paquete
 * @param guestCount - Cantidad de invitados
 * @returns Mensaje formateado
*/
export function getPackageWhatsAppMessage(packageName: string, guestCount: number): string {
  const guestsText = guestCount ? ` para ${guestCount} invitados` : '';
  return `Hola! Me interesa el "${packageName}"${guestsText}. ¿Podrían ofrecerlo?`;
}

```



VARIABLES DE ENTORNO

entre: .env

```
# =====
# LA RESERVA - VARIABLES DE ENTORNO
# =====

# =====
# SUPABASE
# =====

SUPABASE_URL=https://tu-proyecto.supabase.co
SUPABASE_ANON_KEY=tu-clave-anonima-aqui
SUPABASE_SERVICE_KEY=tu-clave-de-servicio-aqui

# =====
# SITIO
# =====

PUBLIC_SITE_URL=http://localhost:3000

# =====
# RESEND (Emails)
# =====

RESEND_API_KEY=re_tu_api_key_aqui

# =====
# WHATSAPP BUSINESS API (Opcional)
# =====

WHATSPAPP_BUSINESS_ID=tu-business-id
WHATSPAPP_ACCESS_TOKEN=tu-access-token

# =====
# GOOGLE MAPS (Opcional)
# =====

PUBLIC_GOOGLE_MAPS_API_KEY=tu-api-key-aqui
```

entre: .env.example

```
# =====
# LA RESERVA - EJEMPLO DE VARIABLES DE ENTORNO
# Copia este archivo a .env y completa los valores
# =====
```

```
# Supabase
SUPABASE_URL=https://tu-proyecto.supabase.co
SUPABASE_ANON_KEY=tu-clave-anonima
SUPABASE_SERVICE_KEY=tu-clave-de-servicio

# Site
PUBLIC_SITE_URL=http://localhost:3000

# Resend (Emails)
RESEND_API_KEY=rexxxxxxxxxxxxx

# WhatsApp Business API
WHATSAPP_BUSINESS_ID=
WHATSAPP_ACCESS_TOKEN=

# Google Maps
PUBLIC_GOOGLE_MAPS_API_KEY=
```

CONFIGURACIONES ADICIONALES

entre: `src/lib/supabase.ts`

```
// src/lib/supabase.ts

import { createClient } from '@supabase/supabase-js';
import type { Database } from '@/types';

const supabaseUrl = import.meta.env.SUPABASE_URL;
const supabaseAnonKey = import.meta.env.SUPABASE_ANON_KEY;

if (!supabaseUrl || !supabaseAnonKey) {
  throw new Error('Missing Supabase environment variables');
}

// Cliente para uso general (cliente y servidor)
export const supabase = createClient<Database>(supabaseUrl, supabaseAnonKey);

// Cliente con service key (SOLO para servidor)
export const supabaseAdmin = createClient<Database>(
  supabaseUrl,
  import.meta.env.SUPABASE_SERVICE_KEY || supabaseAnonKey,
```

```

    }
    auth: {
      autoRefreshToken: false,
      persistSession: false,
    },
  }
);

```

entre: src/lib/resend.ts

```

// src/lib/resend.ts

import { Resend } from 'resend';

const resendApiKey = import.meta.env.RESEND_API_KEY;

if (!resendApiKey) {
  console.warn('Missing RESEND_API_KEY environment variable');
}

export const resend = new Resend(resendApiKey);

/**
 * Envía email de confirmación de cotización
 */
export async function sendQuoteConfirmationEmail(data: {
  to: string;
  name: string;
  eventType: string;
  eventDate: string;
  guestCount: number;
}) {
  try {
    const { data: result, error } = await resend.emails.send({
      from: 'La Reserva <contacto@lareserva.pe>',
      to: data.to,
      subject: 'Cotización Recibida - La Reserva',
      html: `
        <h1>¡Gracias por tu interés, ${data.name}!</h1>
        <p>Hemos recibido tu solicitud de cotización para:</p>
        <ul>
          <li>Tipo de evento: ${data.eventType}</li>
        
```

```

        <li>Fecha: ${data.eventDate}</li>
        <li>Invitados: ${data.guestCount}</li>
    </ul>
    <p>Te contactaremos dentro de las próximas 24 horas.</p>
    <p>Saludos,<br>Equipo La Reserva</p>
    ,
)
;

if (error) {
    console.error('Error sending email:', error);
    return { success: false, error };
}

return { success: true, data: result };
} catch (error) {
    console.error('Error in sendQuoteConfirmationEmail:', error);
    return { success: false, error };
}
}

```

CONSTANTES DE SEO

Agregar al final de constants.ts

```

// =====
// 10. SEO Y METADATA
// =====

/**
 * Metadata por defecto para SEO
 */
export const DEFAULT_SEO = {
    title: 'La Reserva - Mixología Exclusiva',
    description: 'Bartending premium para eventos exclusivos en Lima, Perú.',
    keywords: [
        'bartending lima',
        'mixología perú',
        'eventos premium',
        'cócteles de autor',
        'bartender para bodas',
        'servicio de bar',
    ]
}

```

```
'coctelería lima',
'bartending corporativo',
'eventos exclusivos',
'barra móvil',
],
ogImage: '/images/og-image.jpg',
twitterCard: 'summary_large_image',
} as const;

/***
 * Metadata por página
 */
export const PAGE_METADATA = {
home: {
    title: 'Inicio',
    description: 'Bartending premium para eventos exclusivos en Lima, Per',
},
services: {
    title: 'Servicios de Bartending',
    description: 'Descubre nuestros servicios de bartending y mixología €',
},
packages: {
    title: 'Paquetes de Eventos',
    description: 'Paquetes completos de bartending para bodas, eventos cc',
},
portfolio: {
    title: 'Portafolio de Eventos',
    description: 'Galería de eventos realizados. Cócteles de autor y exp€',
},
about: {
    title: 'Sobre Nosotros',
    description: 'Conoce al equipo de La Reserva. Expertos en mixología c',
},
contact: {
    title: 'Contacto',
    description: 'Contáctanos para tu próximo evento. WhatsApp, email y u',
},
blog: {
    title: 'Blog',
    description: 'Artículos sobre mixología, tendencias y consejos para €',
},
} as const;
```

LISTA DE VERIFICACIÓN FINAL COMPLETA

Archivos Creados

- `src/utils/constants.ts` - Todas las constantes
- `src/utils/validators.ts` - Esquemas de Zod
- `src/utils/formatters.ts` - Funciones de formateo
- `src/utils/utils.ts` - Utilidades generales
- `src/utils/whatsapp.ts` - Ayudantes de WhatsApp
- `src/types/index.ts` - Tipos TypeScript
- `src/lib/supabase.ts` - Cliente Supabase
- `src/lib/resend.ts` - Reenvío del cliente
- `.env` - Variables de entorno
- `.env.example` - no de variables

Configuraciones Verificadas

- Información de contacto actualizada
- URL de redes sociales correctas
- Servicios y paquetes con precios
- Tipos de TypeScript coinciden con BD
- Validadores funcionando
- Formateadores en español
- Ayudantes de WhatsApp operativos
- Variables de entorno configuradas

Pruebas Básicas

```
// Prueba rápida en consola del navegador:  
  
// 1. Formateo de moneda  
import { formatCurrency } from '@/utils/formatters';  
console.log(formatCurrency(1500)); // → 'S/ 1,500'  
  
// 2. Validación de email  
import { isValidEmail } from '@/utils/utils';  
console.log(isValidEmail('test@example.com')); // → true  
  
// 3. WhatsApp URL
```

```
import { getQuoteWhatsAppMessage } from '@/utils/whatsapp';
console.log(getQuoteWhatsAppMessage({
  name: 'Juan',
  eventType: 'Boda',
  eventDate: '2026-06-15',
  guestCount: 100
}));
```

PRÓXIMOS PASOS

Una vez completada toda la configuración:

1. **Verificar importaciones** : Todos los archivos se importan correctamente
 2. **Pruebas** : Probar cada función en desarrollo
 3. **Verificación de tipos** : Ejecutar `pnpm type-check`
 4. **Construir** : Ejecutar `pnpm build` para verificar
 5. **Continuar con** : Desarrollo de componentes y páginas
-

SOPORTE

Si siente problemas:

- Revisa que todos los archivos estén en las ubicaciones correctas.
 - Verifica que las variables de entorno estén configuradas
 - Comprueba que todas las dependencias estén instaladas
 - Ejecuta `pnpm type-check` para ver errores de TypeScript
-