

DOCUMENTACIÓN DEL SISTEMA DE AUTENTICACIÓN OTP

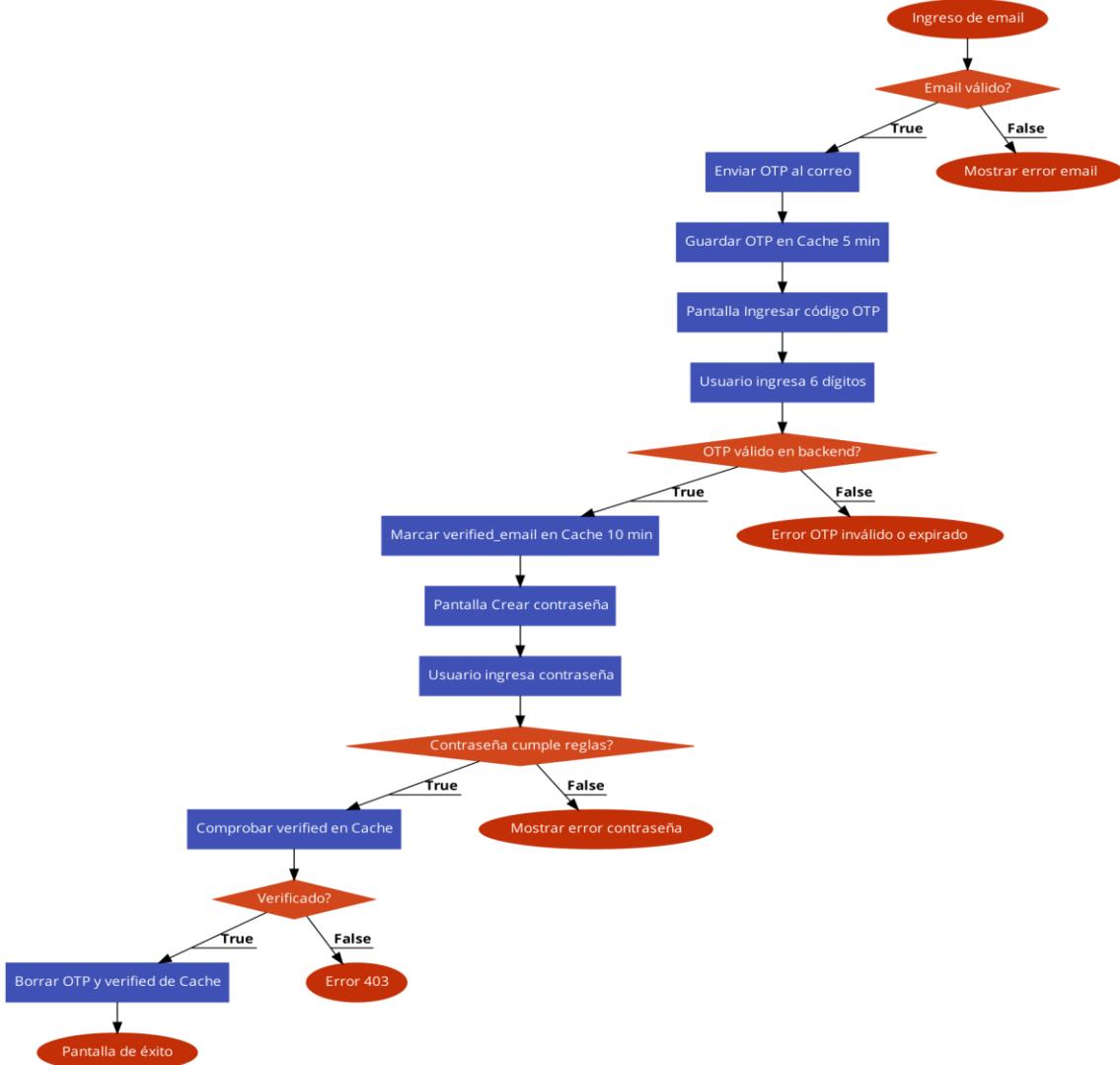
1. Introducción

Este documento describe el flujo lógico y técnico para el sistema de acceso seguro mediante contraseña temporal (OTP) desarrollado en PHP con Laravel. El backend expone una API REST y el frontend está desarrollado en React (Vite).

Stack técnico:

- Backend: Laravel (PHP), Cache, Mail.
- Frontend: React, Vite.
- Almacenamiento temporal: Cache de Laravel.

2. Resumen del Flujo



3. Endpoints

POST /api/send-otp

Función: Generar OTP, guardar en Cache (5 min) y enviar por correo.

POST /api/verify-otp

Función: Validar OTP y marcar email como verificado (10 min).

POST /api/create-password

Función: Validar verificación y limpiar Cache.

4. Estructura Base de Datos

Tabla users (Laravel estándar)

Tabla sessions

Tabla password_reset_tokens

Tabla opcional otp_codes:

```
CREATE TABLE otp_codes (
    id INT AUTO_INCREMENT PRIMARY KEY,
    email VARCHAR(255) NOT NULL,
    code VARCHAR(6) NOT NULL,
    expires_at DATETIME NOT NULL,
    used TINYINT(1) DEFAULT 0,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

5. Consideraciones de Seguridad

- Uso de validaciones Laravel.
- Hash::make() para contraseñas.
- Protección contra fuerza bruta.
- Uso de HTTPS en producción.
- Configuración segura de sesiones.

6. Código de Referencia (Backend)

sendOtp():

```
$otp = rand(100000, 999999);
Cache::put('otp_'.$email, $otp, now()->addMinutes(5));
Mail::raw("Tu código OTP es: $otp", function ($message) use ($email) {
    $message->to($email)->subject('Tu código OTP');
});
```

verifyOtp():

```
$cachedOtp = Cache::get('otp_.'.$request->email);
```

createPassword():

```
Cache::forget('otp_.'.$request->email);
Cache::forget('verified_.'.$request->email);
```

7. Frontend (authService.js)

```
fetch(`${API_URL}/send-otp`, { method: "POST" })
fetch(`${API_URL}/verify-otp`, { method: "POST" })
fetch(`${API_URL}/create-password`, { method: "POST" })
```

8. Resumen Final

Fase 1: Envío OTP → Cache 5 min + Mail

Fase 2: Verificación OTP → Cache 10 min

Fase 3: Crear contraseña → Limpieza Cache