

# ADN Técnico – Corresponsal Bancario: Registro de Retiros

## Escenario

Una organización financiera ha desplegado una red de **corresponsales bancarios** en distintas zonas del país. Estos aliados permiten realizar operaciones básicas de **retiro de efectivo** (💵).

Tu misión es diseñar una **solución tecnológica** que permita registrar estas operaciones, validando un conjunto de reglas de negocio que aseguran el control y buen uso del sistema.


---

## Reglas de Negocio

Asegúrate de aplicar todas estas reglas al procesar cada solicitud de retiro. La solución debe rechazar cualquier operación que no las cumpla.

### 1 Límite diario por corresponsal


 Cada corresponsal tiene un **tope máximo diario** de dinero que puede entregar.

 Si la suma acumulada + el nuevo retiro supera el tope → **rechazar la transacción**.

---

### 2 Monto permitido por transacción


 Cada retiro debe estar entre **\$10.000** y **\$1.000.000** COP.

 Retiros fuera de este rango → **rechazados automáticamente**.

---

### 3 Horario de operación

 Solo se pueden procesar retiros entre **7:00 AM** y **7:00 PM**.

 Fuera de este horario → **no se permiten retiros**.

---

## 4 Bloqueo temporal por errores

⚠ Si un corresponsal falla **3 veces seguidas** al registrar un retiro (por errores en los datos):

🚫 El sistema debe bloquearlo por **10 minutos** antes de permitir una nueva solicitud.

---

## 5 Cierre automático de jornada

📈 Una vez un corresponsal registre **más de 100 operaciones** en un mismo día:

🛑 Se considera que ha cerrado su jornada y no puede registrar más retiros hasta el día siguiente.

---



## Datos esperados en la solicitud de retiro

La operación debe recibir y procesar al menos los siguientes campos:

- 🆔 ID del corresponsal
  - 💵 Monto del retiro
  - 📅 Fecha y hora del intento
  - 📱 Canal de atención (ejemplo: ventanilla, dispositivo móvil)
- 



## Stack tecnológico a utilizar

Tecnología	Uso esperado
🐍 Python	Backend y lógica
⚛️ React	Frontend
🐘 PostgreSQL	Persistencia
☁ AWS	Despliegue