



# SOP: Deploying a Cloud-Based N8N Instance Using Render and Supabase

---

## Overview

This SOP details how to deploy an N8N automation platform instance on the cloud using Render (as the hosting provider) and Supabase (for database storage). By the end of this process, you'll have a live N8N cloud instance with persistent storage, webhook accessibility, and easy migration capability between different hosts using consistent environmental variables.

---

## Step-by-Step Guide

### 1. Create a Supabase Account and Project

- Go to <https://supabase.com> and sign up using your email.
- Create a new organization and set the pricing to "Free".
- Choose a region close to you (e.g., **West Europe**).
- Set a **database password** and save it securely.
- Create a new project within the organization.

### 2. Create an Account on Render

- Visit <https://render.com> and sign up.
- Click **New Web Service**.
- Select **Deploy an existing image**.
- Use Docker image:

n8nio/n8n

- e. Set service name (e.g., `n8ntutorial`) and choose the closest region.
- f. Select the `Free` plan for now (upgrade later if needed).

### 3. Add Environment Variables in Render

- Set the following variables (update with your actual values from Supabase):

```
N8N_PROTOCOL=https
N8N_WEBHOOK_URL= (leave blank for now)
DB_TYPE=postgresdb
DB_POSTGRESDB_DATABASE=postgres
DB_POSTGRESDB_HOST=<Supabase Host>
DB_POSTGRESDB_PORT=<Supabase Port>
DB_POSTGRESDB_SCHEMA=public
DB_POSTGRESDB_USER=<Supabase User>
DB_POSTGRESDB_PASSWORD=<your saved password>
GENERIC_TIMEZONE=Europe/London
N8N_ENCRYPTION_KEY=<generated 256-bit key>
TZ=Europe/London
```

1. Retrieve host, port, user from Supabase → `Project` → `Connect` → use Transaction Pooler section.
2. Generate a 256-bit key from <https://acte.ltd/utils/randomkeygen>.

### 4. Deploy the Web Service

- a. Click `Deploy Web Service`.
- b. Wait until deployment is complete and a public Render URL is generated.
- c. Save this URL.

### 5. Update Webhook Environment Variable

- a. Copy the Render URL.
- b. Go to Render → Service → Environment → Edit.
- c. Set:

```
N8N_WEBHOOK_URL=https://your-render-url.onrender.com
```

- d. Save and redeploy the service.

## 6. Register and Initialize N8N

- a. Visit the Render URL in your browser.
- b. Complete registration with:
  - Email
  - First and last name
  - Password
- c. Retrieve your license key from your email and activate it in the N8N settings.

## 7. Test a Webhook Workflow

- a. In N8N, create a new workflow using a **Webhook** trigger.
- b. Copy the webhook URL.
- c. Go to <https://postman.com> and send a **GET** or **POST** request to the URL.
- d. Confirm the workflow is triggered and data is received.

## 8. Test HTTP Request Node

- a. In your N8N workflow, add an HTTP request node.
- b. Set the URL to **<https://google.com>**.
- c. Execute the node and confirm HTML is returned.

## 9. Backup and Migration Readiness

- a. All workflows and credentials are stored in Supabase.
- b. You can redeploy to any other host (e.g., Hostinger, LSTU) by using the same environment variables.
- c. This ensures continuity if your current host goes down or needs maintenance.

---

## Warnings and Notes

- **Save Database Credentials:** The Supabase password and generated encryption key must be stored safely. They're required for redeployment.
- **Delay in Activation:** Free Render instances may go idle after inactivity. Revisit the link to wake them up.
- **Webhook Update Required:** You must update `N8N_WEBHOOK_URL` after the initial deployment to ensure external integrations work.

## Visual Diagram

flowchart TD

```
A[Create Supabase Account] → B[Create Project & Save DB Info]
B → C[Create Render Account]
C → D[Deploy Docker Image on Render]
D → E[Set Env Variables from Supabase]
E → F[Deploy Web Service]
F → G[Copy Public Render URL]
G → H[Update Webhook URL in Env]
H → I[Re-deploy Render Service]
I → J[Open N8N & Register]
J → K[Test Webhook with Postman]
K → L[Test HTTP Request to Google]
L → M[Backup Ready with Supabase]
```