

package.md

05-technical-path/supabase-n8n-technical-path.md

PS> boring fix lead-followup

status: practical, transparent, no hype

Supabase / n8n Technical Path

This path is for custom websites, custom CRMs, technical founders, or teams that want a real backend.

terminal

Custom form or Webflow form

→ Supabase

→ n8n

→ Resend / Postmark / Twilio

What you will build

A database-backed follow-up system where new leads are stored in Supabase, n8n handles automation, and email/SMS providers send notifications or responses.

Supabase schema draft

terminal

```
create table leads (  
  id uuid primary key default gen_random_uuid(),  
  created_at timestamptz not null default now(),  
  updated_at timestamptz not null default now(),  
  lead_name text not null,  
  business_name text,  
  email text,  
  phone text,  
  source text,  
  service_requested text,  
  original_message text,  
  status text not null default 'new',  
  priority text not null default 'normal',  
  last_contacted_at timestamptz,  
  next_follow_up_at timestamptz,  
  assigned_owner text,  
  missing_info text,  
  ai_summary text,  
  outcome text,  
  notes text  
);  
  
create table follow_ups (  
  id uuid primary key default gen_random_uuid(),  
  created_at timestamptz not null default now(),  
  lead_id uuid references leads(id) on delete cascade,  
  due_at timestamptz not null,  
  completed_at timestamptz,  
  type text not null default 'manual',  
  notes text  
);  
  
create table messages (  
  id uuid primary key default gen_random_uuid(),  
  created_at timestamptz not null default now(),  
  lead_id uuid references leads(id) on delete cascade,  
  direction text not null,  
  channel text not null,  
  subject text,  
  body text,
```

```
provider_message_id text
);
```

RLS note

If this is used in production, do not expose service-role credentials to the browser.

Use:

```
terminal
```

```
server-side API route
Supabase Edge Function
n8n credential store
```

Never:

```
terminal
```

```
service role key in frontend JavaScript
```

n8n workflow blueprint 1 – New lead

```
terminal
```

```
Webhook receives lead
→ Validate required fields
→ Insert lead into Supabase
→ Send confirmation email via Resend/Postmark
→ Send internal notification
→ Create follow-up record due tomorrow
```

n8n workflow blueprint 2 – Daily follow-up reminder

```
terminal
```

```
Cron: every weekday at 9 AM
→ Query Supabase for follow_ups where due_at <= now and completed_at is null
→ Send summary email/slack message to owner
→ Include lead name, service requested, last contacted date, and next action
```

n8n workflow blueprint 3 – SMS response optional

terminal

```
Webhook receives lead
→ Check phone exists
→ Send short SMS via Twilio/OpenPhone if consent/use case allows
→ Log message in messages table
```

Email provider options

Tool	Use case
Resend	Clean developer-friendly transactional email
Postmark	Strong transactional email reliability
Mailgun	Flexible developer email platform
SendGrid	Common enterprise-ish email option

SMS provider options

Tool	Use case
Twilio	Flexible programmable SMS
OpenPhone	SMB-friendly phone/SMS workflow
Aircall/Dialpad	Teams with call-center style workflows

Test plan

- Submit a webhook test payload.
- Confirm Supabase insert.
- Confirm email sends.
- Confirm follow-up record exists.
- Confirm message log exists.
- Confirm failed inserts are logged.
- Confirm no secret keys are exposed in the browser.