

MINI PACK

Signup Emails That Actually Arrive

Fix Supabase auth emails that never show up or
land in spam.

For Cursor · Claude Code · Claude · ChatGPT · Windsurf · Lovable · Bolt · v0 · Replit

9 prompts · 4 sections

Welcome! How to Use This Pack

Someone signs up for your app and waits for the confirmation email. It never comes. Or it lands in spam, or the link inside says it has expired before they even clicked it.

This pack fixes that for apps that use **Supabase Auth**. The most common cause is simple: Supabase's built-in email sender is only meant for testing. Unless you set up your own email service (custom SMTP), it only delivers to members of your Supabase team, and it can send very few emails per hour. Real users never get anything.

The prompts walk you through finding the cause, connecting a real email service, staying out of spam, and fixing the links inside the emails. Supabase's dashboard changes over time, so the prompts ask your AI to say when it isn't sure. If what your AI says doesn't match your screen, trust the Supabase dashboard and the official docs at supabase.com/docs.

How each prompt is laid out

- **Title:** what the prompt is.
- **Use this when...:** one or two sentences on what the prompt is for and when to reach for it.
- **Fill in:** the blanks to replace before you paste.
- **The prompt:** the box you copy.
- **Tips to customize:** optional ways to change it.

Anything in **[BRACKETS IN CAPS]** is a blank to fill in. **Never paste real passwords, SMTP passwords, or API keys into an AI chat.** Use names like `SMTP_PASSWORD` instead.

Works best with: Agentic editors like Cursor and Claude Code can read your auth code and email templates for you. App builders like Lovable, Bolt, and v0 work well when your app is connected to your own Supabase project. Every prompt also works in any AI chat.

On Lovable Cloud instead of your own Supabase project? The Supabase dashboard steps in this pack won't match your screen. Lovable has its own email setup for Cloud apps, so check Lovable's docs on sending emails from your own domain.

Using Ship It Live or Ship It Safe too? Ship It Live 5.1 covers login redirects that send people to localhost after you deploy, and Ship It Safe 5.3 covers stopping bots and spam sign ups. This pack covers getting the emails delivered.

Quick Start: Find Your Problem

- **Nobody gets your signup emails except you?** Prompt 1.1, then 2.1.
- **Emails arrive, but in spam?** Prompts 3.1 and 3.2.
- **The link in the email says "expired" or "invalid"?** Prompt 4.1.
- **Password reset email arrives, but the reset doesn't work?** Prompt 4.2.

Golden rule: Supabase's built-in email sender is for testing only. Before real users sign up, connect your own email service with custom SMTP.

What's Inside

1. Find Out Why Emails Don't Arrive (2 prompts)
2. Connect a Real Email Service (2 prompts)
3. Stay Out of Spam (3 prompts)
4. Fix the Links Inside the Email (2 prompts)

SECTION 1

Find Out Why Emails Don't Arrive

Work out where the email is getting stuck before you change anything.

1.1 "My Confirmation Email Never Arrives"

Use this when people sign up but never get the confirmation, magic link, or reset email, and you're not sure why. It's for checking the likely causes in order, starting with the built-in sender's limits, before you change any settings.

Fill in: [SKILL LEVEL], [TECH STACK], [WHICH EMAIL] (e.g. signup confirmation, magic link, password reset, invite), [WHO DOESN'T GET IT] (e.g. "everyone except me", "only Outlook users", "nobody"), [CUSTOM SMTP] (yes / no / not sure), [WHAT I SEE] (e.g. "no error at all", "Email address not authorized", "429 Too Many Requests").

Users of my app aren't getting their Supabase Auth emails. Help me find out why, step by step. I'm a [SKILL LEVEL], so explain each check in plain English.

Tech stack: [TECH STACK]

Which email: [WHICH EMAIL]

Who doesn't get it: [WHO DOESN'T GET IT]

Have I set up custom SMTP? [CUSTOM SMTP]

What I see: [WHAT I SEE]

Please check these in order:

1. Am I still using Supabase's built-in email sender? Explain that without custom SMTP, Supabase only delivers to addresses that are members of my project's organization team, and other addresses fail with "Email address not authorized". Tell me where to check this in the dashboard.
2. Could I be hitting the email rate limit? The built-in sender allows very few emails per hour, and Supabase also limits how often the same user can request another email. Explain what a "429 Too Many Requests" error means here.
3. Is the email actually being sent? Show me how to find the matching events in my project's Auth logs (Logs, with the Auth log type selected) and what a failed send looks like.
4. Is email confirmation turned on for my project, and is my code calling the right function (signUp, signInWithOtp, resetPasswordForEmail)?
5. For password resets: remind me that Supabase doesn't send a reset email when no account exists for that address, but still returns no error.
6. Could the email be arriving but going to spam or a promotions tab? Tell me what to ask the user to check.

Then tell me the most likely cause, the exact fix, and how to confirm it worked.

If you're not sure about a dashboard location, say so and point me to the Supabase docs page.

 **Tips to customize:** If you only ever tested with your own email address, that's the classic sign: you're on your Supabase team, so the built-in sender delivers to you and to nobody else.

1.2 Read the Auth Logs Like a Detective

Use this when you need proof of what happened to one specific email, instead of guessing. It's for finding and reading the Auth log entries for one signup or reset attempt so you know whether Supabase tried to send it and what went wrong.

Fill in: [WHEN I TRIED] (e.g. "about 3:15 PM today"), [TEST EMAIL ADDRESS DOMAIN] (e.g. "a gmail.com address"), [WHAT I COPIED FROM THE LOGS].

Help me read my Supabase Auth logs to find out what happened to one auth email.

When I tried: [WHEN I TRIED]

The address was: [TEST EMAIL ADDRESS DOMAIN] (I'm not sharing the full address)

Log entries I copied (with personal details removed): [WHAT I COPIED FROM THE LOGS]

Please:

1. Tell me how to open Logs in the Supabase dashboard, select the Auth log type, set the time range, and search the event messages.
2. Explain each log entry I pasted in plain English.
3. Tell me whether Supabase tried to send the email, refused it, or hit a limit.
4. If the log shows an SMTP error from my email service, explain what that error usually means and what to check in that service's dashboard.
5. Tell me the next step to fix it.

If the logs I pasted aren't enough to tell, say exactly what else to look for.

 **Tips to customize:** Remove email addresses and user IDs before you paste logs into an AI chat. A domain like "gmail.com" is usually enough for the AI to work with.

SECTION 2

Connect a Real Email Service

Replace the built-in test sender with one that can email real users.

2.1 Set Up Custom SMTP Step by Step

Use this when you're ready for real users to receive your auth emails, or Prompt 1.1 showed you're still on the built-in sender. It's for choosing an email service and connecting it to Supabase Auth with custom SMTP, without pasting any secrets into chat.

Fill in: [EMAIL SERVICE] (e.g. Resend, Postmark, AWS SES, Brevo, or "help me choose"), [MY DOMAIN] (e.g. myapp.com), [SENDER ADDRESS] (e.g. no-reply@myapp.com), [SENDER NAME] (e.g. your app's name), [WHERE DNS IS MANAGED] (e.g. Cloudflare, Namecheap, GoDaddy).

```
Help me set up custom SMTP for Supabase Auth so my signup, magic link, and password reset emails reach real users.
```

```
Email service: [EMAIL SERVICE]
My domain: [MY DOMAIN]
Sender address I want: [SENDER ADDRESS]
Sender name: [SENDER NAME]
Where my DNS is managed: [WHERE DNS IS MANAGED]
```

```
Please walk me through:
```

1. If I asked for help choosing: compare 2-3 services that support SMTP and work with Supabase, in plain English. Tell me to check their current pricing and free tiers on their own sites.
2. Setting up my sending domain in that service and adding the DNS records it gives me (I'll copy them from the service; explain what each one is for).
3. Finding the SMTP settings in that service: host, port, username, and password. Tell me which of these is a secret, and remind me to paste it only into the Supabase dashboard, never into chat or code.
4. Where to turn on custom SMTP in my Supabase project's Authentication settings, and which fields to fill in (sender email, sender name, host, port, username, password).
5. How to send one test (for example, a password reset to an address that's not on my Supabase team) and how to confirm it went through my email service.
6. What to do if the test fails: which error messages mean wrong credentials, wrong port, or an unverified domain.

```
If you're not sure about a setting name or location, say so and tell me which docs page to check.
```

 **Tips to customize:** Using Resend? Its docs have a step-by-step page for Supabase SMTP. The service's own setup page is always the best source for host, port, and username.

2.2 Raise the Email Rate Limit Safely

Use this when custom SMTP works, but some users still don't get emails during busy moments, or you're planning a launch. It's for setting an email sending limit that fits your real signup volume without opening the door to abuse.

Fill in: [EXPECTED SIGNUPS] (e.g. "about 20 a day", "a few hundred on launch day"), [EMAIL SERVICE], [WHAT HAPPENS NOW] (e.g. "some users get a 429 error", "worried about launch day").

I've set up custom SMTP for Supabase Auth. Help me set the right email rate limit for my project.

Expected signups: [EXPECTED SIGNUPS]

Email service: [EMAIL SERVICE]

What happens now: [WHAT HAPPENS NOW]

Please:

1. Explain in plain English that Supabase sets a low starting email limit after you switch on custom SMTP, to protect a brand-new sending setup, and that I can change it on the Rate Limits page in Authentication.
2. Help me estimate how many auth emails I send per hour (signups, magic links, resets, email changes), including a busy launch hour.
3. Suggest a limit with some headroom, and explain the trade-off: too low blocks real users, too high lets bots burn through my email service's quota and reputation.
4. Remind me to check my email service's own sending limits so the two match.
5. Tell me what users see when they hit the limit, and suggest a friendly message for my app to show instead of a raw error.
6. Suggest protections to add before raising the limit a lot, like CAPTCHA on the signup form.

 **Tips to customize:** Expecting a big launch? Supabase suggests telling your email service ahead of time, since sudden spikes can hurt your sending reputation. For bot protection, see Ship It Safe 5.3.

SECTION 3

Stay Out of Spam

Make inbox providers trust your emails.

3.1 SPF, DKIM, and DMARC in Plain Words

Use this when your emails arrive but land in spam, or your email service says your domain isn't fully set up. It's for understanding and checking the three DNS records that prove your emails really come from you.

Fill in: [MY DOMAIN], [EMAIL SERVICE], [WHERE DNS IS MANAGED], [MY CURRENT DNS RECORDS] (copy the type, name, and value of any TXT, CNAME, or MX records; no secrets).

My app's auth emails are landing in spam. Help me check my email authentication records in plain words.

Sending domain: [MY DOMAIN]

Email service: [EMAIL SERVICE]

Where my DNS is managed: [WHERE DNS IS MANAGED]

My current DNS records: [MY CURRENT DNS RECORDS]

Please:

1. Explain SPF, DKIM, and DMARC like I'm new to this: what each one proves and what happens when it's missing.
2. Compare my records with what my email service asks for, and list anything missing, duplicated, or wrong. (A domain should have only one SPF record.)
3. Tell me exactly which record to add or change, and where, at my DNS provider.
4. Suggest a gentle starting DMARC policy for a small app, and explain what it does.
5. Explain how long DNS changes can take and how to check that they're live.
6. Remind me to look for a "verified" status in my email service before testing again.

If any value depends on my email service, tell me to copy it from that service's dashboard instead of guessing.



Tips to customize: Supabase also suggests using a separate sending domain (or subdomain) for auth emails, like auth.myapp.com, so marketing emails can't hurt their reputation. Ask: "Should I send auth emails from a subdomain? Walk me through the trade-offs."

3.2 Rewrite My Auth Emails So They Don't Look Like Spam

Use this when your confirmation or reset emails look like marketing, still use the default text, or keep landing in spam. It's for rewriting your Supabase email templates so they're short, clear, branded, and still work.

Fill in: [APP NAME], [WHICH TEMPLATES] (e.g. confirm sign up, magic link, reset password), [PASTE THE CURRENT TEMPLATE], [TONE] (e.g. "friendly and short").

Rewrite my Supabase Auth email templates so they look trustworthy and are less likely to be flagged as spam.

App name: [APP NAME]

Templates to rewrite: [WHICH TEMPLATES]

Current template: [PASTE THE CURRENT TEMPLATE]

Tone: [TONE]

Rules:

1. Keep every template variable that makes the email work, like `{{ .ConfirmationURL }}`, `{{ .Token }}`, or `{{ .TokenHash }}`, exactly as written. Tell me which ones each template needs.
2. Short, plain subject lines with no emojis, like "Confirm your email" or "Reset your password".
3. One clear button or link, and as few other links and images as possible.
4. No marketing, taglines, or promotions, and don't describe what my app is in detail.
5. Explain why the email was sent and what to do if the person didn't ask for it.
6. Simple HTML that works in most email clients.

Then tell me where to paste each template in the Supabase dashboard (Email Templates), and remind me to send myself a test after saving.



Tips to customize: Supabase recommends changing auth templates rarely, and in one bigger update rather than lots of small ones. If your email service has link or open tracking, consider turning it off for auth emails, because tracking can rewrite the links in Supabase's templates.

3.3 Test Delivery Before Real Users Sign Up

Use this when you've set up custom SMTP and DNS and want to be sure emails land in the inbox before you launch. It's for running a short, careful delivery test across a few inbox providers without hurting your new domain's reputation.

Fill in: [EMAIL SERVICE], [TEST INBOXES] (e.g. "a Gmail and an Outlook account I own"), [WHICH EMAILS] (e.g. signup confirmation and password reset).

```
Help me test my Supabase Auth emails before launch, carefully, so I don't damage
my new sending domain.
```

```
Email service: [EMAIL SERVICE]
Test inboxes I own: [TEST INBOXES]
Emails to test: [WHICH EMAILS]
```

```
Please give me a short test plan:
```

1. Send each email type once to each test inbox. Keep the total small, because lots of test emails from a new domain can look suspicious.
2. For each one, record: did it arrive, which folder it landed in, and how long it took.
3. Show me how to open the full message headers in common mail apps and check whether SPF, DKIM, and DMARC passed. Tell me what "pass" and "fail" look like.
4. Click each link once and confirm it lands on the right page of my live app.
5. Check my email service's dashboard for each send: delivered, bounced, or rejected.
6. If anything fails, tell me which prompt or setting to look at next.

```
Only use addresses I own. Never test with made-up addresses, because bounces hurt
deliverability.
```



Tips to customize: Testing on your computer with the Supabase CLI? Local emails are caught by a local inbox tool (Mailpit) instead of being sent. Run `supabase status` to find its URL.

SECTION 4

Fix the Links Inside the Email

The email arrived. Now make the link work.

4.1 "Token Has Expired or Is Invalid"

Use this when people click the link in your confirmation or magic link email and see "Token has expired or is invalid", often on work or school email. It's for finding out whether something used the link before the person did, and switching to a link or code that survives it.

Fill in: [TECH STACK], [WHICH EMAIL], [WHO SEES THE ERROR] (e.g. "only people with work Outlook accounts", "everyone"), [PASTE THE TEMPLATE AND THE PAGE THE LINK OPENS].

Users click the link in my Supabase Auth email and get "Token has expired or is invalid". Help me find out why and fix it.

Tech stack: [TECH STACK]

Which email: [WHICH EMAIL]

Who sees the error: [WHO SEES THE ERROR]

My template and the page the link opens: [PASTE THE TEMPLATE AND THE PAGE THE LINK OPENS]

Please check:

1. Link prefetching: some email security tools open links in incoming emails to scan them, which can use up a one-time link before the person clicks. Explain how to tell if this is happening.
2. Whether my email service's link tracking is rewriting the link.
3. Whether the person clicked an older email after requesting a newer one, or waited too long.
4. Whether my app's confirm page handles the link the way my setup expects.

Then give me the fix that fits my app, explained simply. Supabase's docs describe options such as sending a one-time code ({{ .Token }}) that the user types in, or linking to a page with a button that completes the confirmation. Show me the template change and the code change, and tell me what to test afterward.



Tips to customize: If the link opens localhost or the wrong site instead of showing an error, that's a redirect problem. Ship It Live 5.1 covers it.

4.2 Password Reset That Works End to End

Use this when the reset email arrives, but clicking it doesn't let people set a new password, lands them on the wrong page, or logs them in without asking for one. It's for tracing the whole reset flow, from the email request to saving the new password, and fixing the broken step.

Fill in: [TECH STACK], [MY LIVE URL], [MY RESET PAGE] (e.g. /account/update-password), [WHAT HAPPENS] (e.g. "lands on the homepage", "the form says I'm not logged in"), [PASTE OR TAG THE RESET-RELATED FILES].

My Supabase password reset flow is broken. Help me trace it step by step and fix the broken part.

Tech stack: [TECH STACK]

Live site: [MY LIVE URL]

Page where users should set a new password: [MY RESET PAGE]

What happens: [WHAT HAPPENS]

Relevant files: [PASTE OR TAG THE RESET-RELATED FILES]

Please check each step:

1. The request: does my code call `resetPasswordForEmail` with a `redirectTo` that points to my reset page?
2. The allow list: is that exact URL allowed in my Supabase Redirect URLs, and is my Site URL set to my live site instead of localhost?
3. The template: if I pass `redirectTo`, does my reset template use `{{ .RedirectTo }}` where it needs to, instead of only `{{ .SiteURL }}`?
4. The landing: when the user arrives, does my app create the session correctly for my setup (client-only or server-side)? Explain which flow I'm using.
5. The update: does the reset page call `updateUser` with the new password once the user has a session, and show a clear success or error message?
6. Edge cases: an expired link, a reused link, or a reset requested for an address with no account.

Show me the smallest code or settings change for the broken step, then a short test plan using a real address I own.



Tips to customize: For server-rendered apps, Supabase's password guide shows a reset template that links to a confirm route in your app. Ask: "Is my app using the PKCE flow? If so, show me the matching template and confirm route."

Bonus: Magic Phrases for Auth Emails

- "Am I still on Supabase's built-in email sender?"
- "Check the Auth logs before changing any settings."
- "Keep every template variable exactly as it is."
- "Never put the SMTP password in my code or this chat."
- "Check supabase.com/docs and tell me if you're not sure."
- "Give me a test plan using only addresses I own."

Signup Emails That Actually Arrive

That's the whole pack: **9 prompts across 4 sections**. Find where the email gets stuck, connect a real email service with custom SMTP, set up SPF, DKIM, and DMARC, and make sure every link lands where it should.

Happy launching!