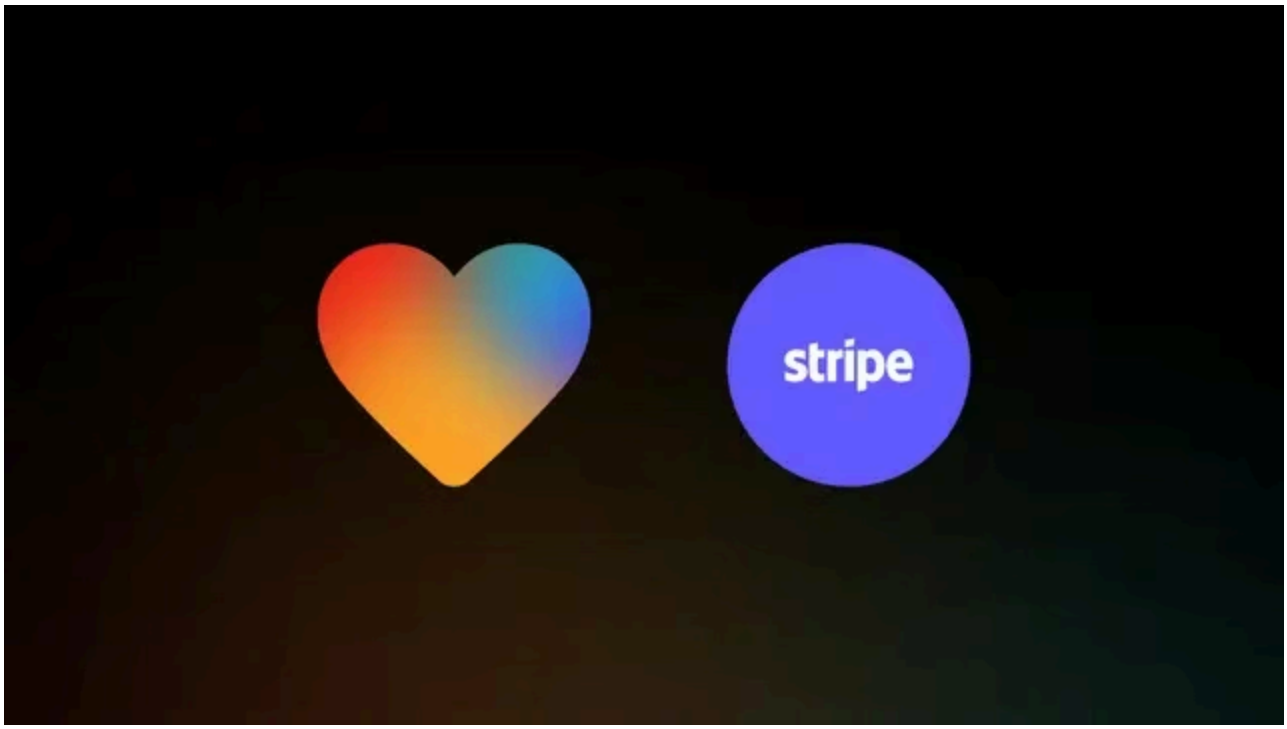
Integrations

Stripe integration

Copy page

How to set up payments in your app using our Stripe integration



Lovable now lets you set up Stripe entirely through **chat**.

Chat-driven auto-setup (recommended)

After you connect **Supabase** and save your **Stripe Secret Key** via **Add API Key**, just describe what you need:

> "Add three subscription tiers ..."
>
> "Create a one-time checkout for my e-book at $29"
>
> Lovable generates the checkout / portal edge functions, database tables with RLS, and UI buttons—no manual coding or webhooks unless you ask for them.

Lovable

›

Key takeaways

Use the **chat-driven flow** for both subscriptions and one-off payments.

Never paste your Stripe Secret Key in chat. Configure it via the in-chat **Add API Key** form.

Webhooks are opt-in. Lovable relies on edge-function polling unless you request webhooks.

Debug in **Browser Console → Network/Errors**, **Supabase → Edge Functions → Logs**, and **Stripe Dashboard → Logs**.

Always test in **Stripe Test Mode**, then deploy.

Requirements

Before integrating Stripe, ensure the following prerequisites are met:

The project **must** be connected to Supabase. **Learn more about Supabase**

A **Stripe account** with properly configured products.

A working **frontend and backend**:

For individual product sales, ensure a shopping cart and checkout page are functional.

For subscriptions, set up login functionalities and different pricing tiers.

Please note Stripe integration doesn't work in preview. To test the integration, make sure to deploy. You should also make sure to be in test mode in Stripe when trying out the functionality. When testing payment, card number: 4242 4242 4242 4242, any 3 digits as CVC and any future date will work as a card.

Lovable

›

Lovable what you need in chat — no manual Payment Links required.

1 **Step 1**

Prep your project

Supabase connected

Stripe Secret Key added via the in-chat **Add API Key** form

(Optional) Prices or product IDs handy

2 **Step 2**

Examples:

Create a one-time checkout for my “Digital Course” at $29

Set up an annual Premium plan for $99, tied to each user’s id

3 **Step 3**

Review & apply Lovable auto-scaffolds the Edge Functions, database tables, and UI components (all tied to the user’s id in Supabase). Check the preview, then click **Apply** to deploy.

Subscriptions should always linked to the authenticated user’s `id` in Supabase for secure, role-based access.

Advanced integration: Webhooks & Supabase

Lovable

›

The Edge Function that handles the necessary changes to the user account should be set up automatically by the AI.

1 Connect Supabase to Your Project

Getting started is simple. Lovable makes connecting Supabase effortless with a built-in native integration:

1. Click the **Supabase button** in the top right corner of Lovable.
2. Follow the instructions to link your project.
3. Once connected, Supabase enables secure payment processing, subscription management, webhook handling, customer data storage, and error handling.

2 Secure Payment Processing

Initiate the process by prompting Lovable:

Let's connect Stripe to my project. We will begin with secure payment processing.

Lovable will generate the necessary SQL schema for handling payments. This includes database tables for users, subscriptions, and payments. You can review and customize these tables to fit your specific product needs before applying changes.

3 Implement Edge Functions for Webhooks

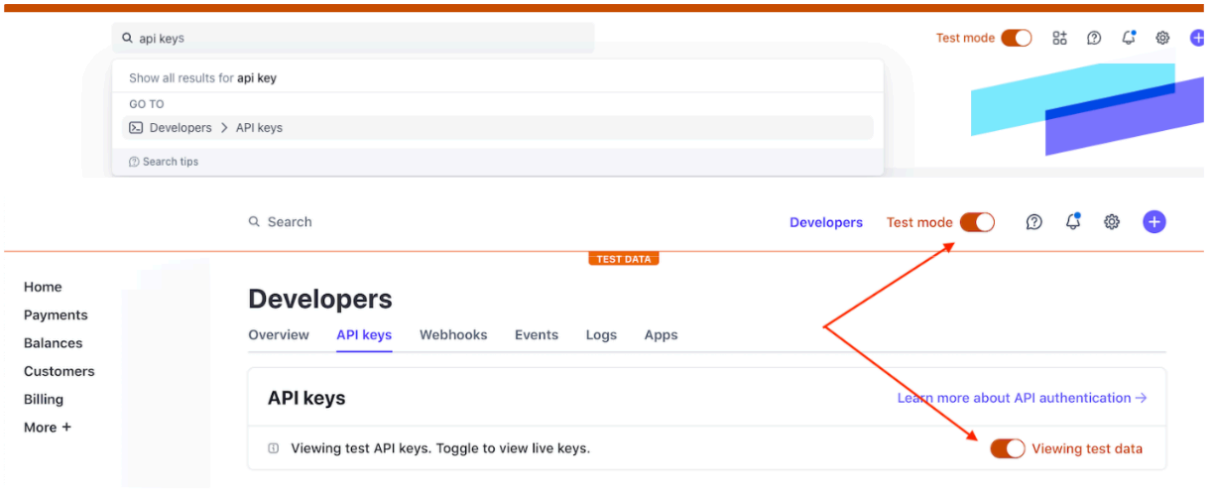
Edge Functions in Supabase act as small, high-performance serverless functions that run close to the user, ensuring fast responses. They help process webhook events, such as payment confirmations, before updating the database.

Lovable

›

2 Step 2

Go to Stripe Dashboard > Developers > Webhooks > Create an Event Destination.



3 Step 3

Select Webhook Events that align with your project needs:

```
payment_intent.succeeded

payment_intent.payment_failed

customer.subscription.created

customer.subscription.updated

customer.subscription.deleted
```

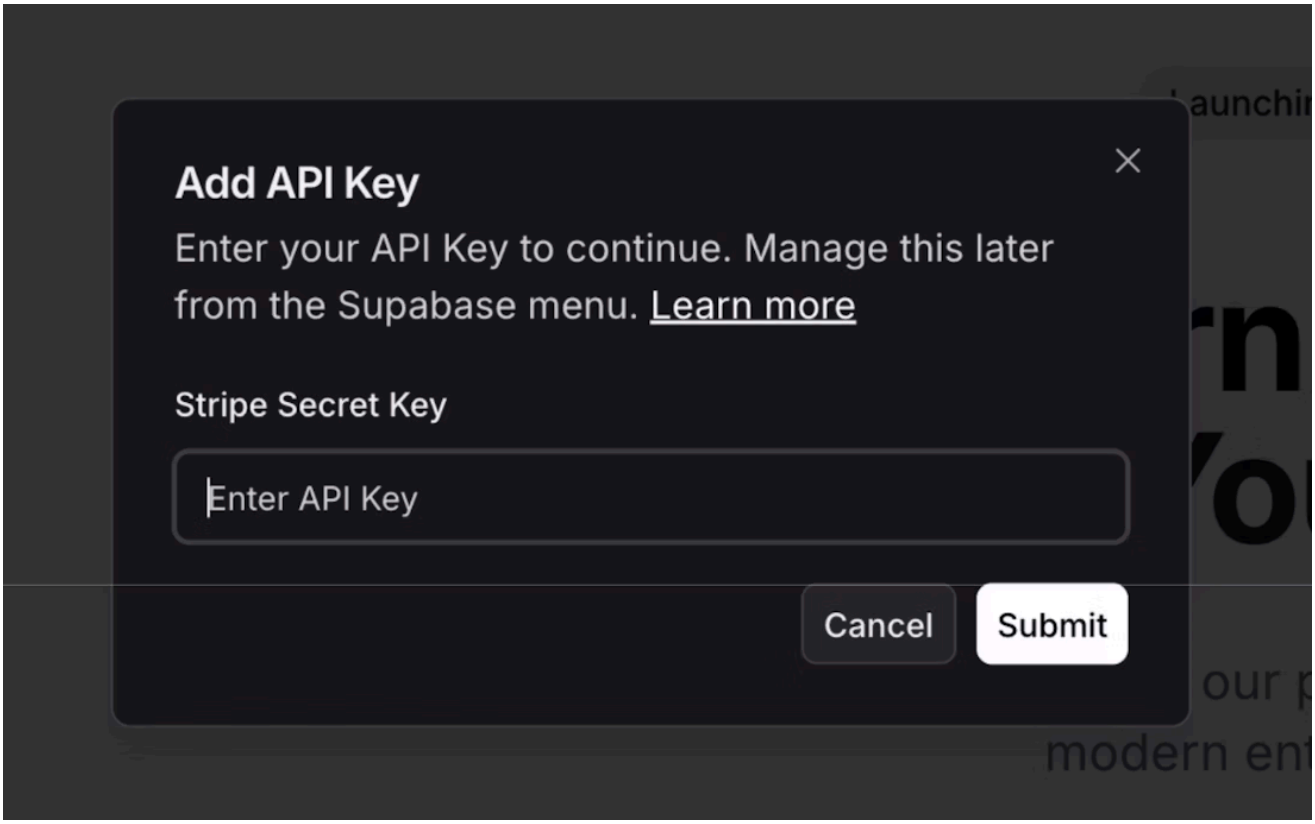
4 Step 4

Enter the Endpoint URL from Supabase.

If unsure about naming the secret, ask Lovable in **chat mode** for guidance.

4 Securely Add API Keys

To integrate Stripe securely, avoid sharing your API key directly in chat. Instead:



1 Step 1

Go to Stripe Dashboard > Developers > API Keys.

2 Step 2

Lovable

›

Use the Lovable "Add API Key" feature to securely store it.

> ⚠ **Important Security Warning**
>
> Never paste your **Secret Key** directly in Lovable chat. Treat it like the keys to your house—exposing it could allow unauthorized access to your Stripe account. Instead, store it securely using Lovable's API key feature.

5 Testing Your Integration

Use Stripe's Test Mode to safely test payments.

Test card details:

Card Number: `4242 4242 4242 4242`

Any future expiration date

Any 3-digit CVC

Deploy your app—Stripe integration does not work in preview mode.

Debugging & Troubleshooting

Check Console Logs
Review Supabase Logs
Verify Webhook Events in Stripe
Use Lovable Chat Mode

Supabase integration

Shopify integration