



Idea to app in seconds.

Lovable is your superhuman full stack engineer.

Ask Lovable to create a dashboa

Attach Import

Public

Personal website ↑

Task manager ↑

Bill splitter ↑

Hacker News top 100 ↑

Tuto

Créer sa première application avec Lovable.dev

Ce tutoriel vous est offert par 24pm Academy et Neodia

- ✓ Formations
- ✓ Coaching
- ✓ Applications IA
- ✓ Stratégie IA pour Comex

Raphaël Richard



Qualiopi
processus certifié
RÉPUBLIQUE FRANÇAISE



Besoin de former vos
équipes sur l'IA
et l'IA générative et de
développer des applis d'IA

Nous sommes là

Neodia
24pm Academy

Formations IA
Conseil IA
AMAO IA

📞 06 62 55 00 10

📱 Raphaël Richard

✉️ Raphael.richard@neodia.com

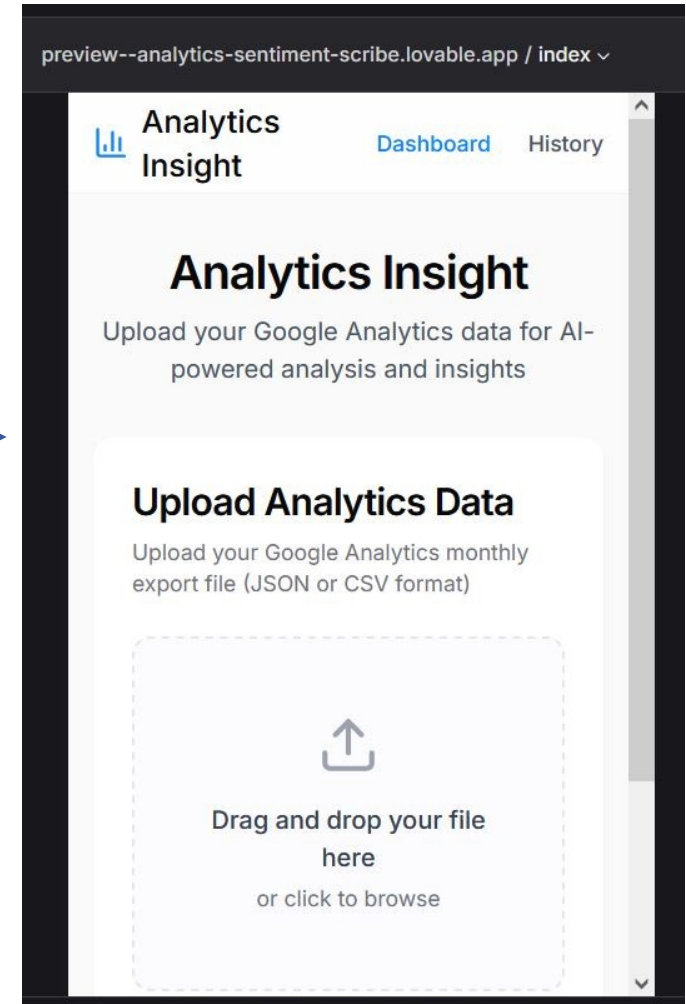
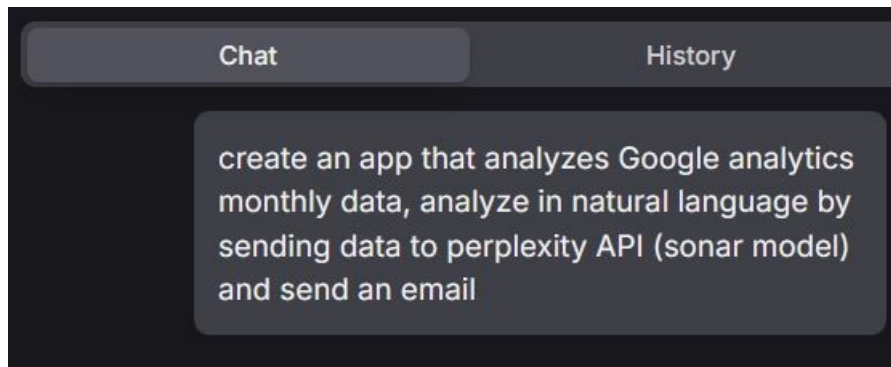
🌐 <https://24pm.Com>

Qu'est-ce lovable.dev ?

C'est un générateur d'application sur mesure

- ✓ Vous décrivez votre application et ses fonctionnalités
- ✓ Lovable définit l'architecture technique et génère petit à petit tout le code nécessaire
- ✓ Lovable vous permet de tester l'application via une interface web ou mobile
- ✓ Vous demandez les corrections nécessaires
- ✓ Une fois qu'elle est finalisée, vous téléchargez le code pour l'installer l'application sur vos serveurs ou vous publiez cette application directement sur Lovable qui peut l'héberger

Un prompt, une application



Cela peut être aussi simple que cela !

Est-ce magique ?

Non, car il existe de nombreuses limitations

- ✓ Lovable ne code pas les applications très complexes
- ✓ Lovable bugue parfois au moment de la génération de code
- ✓ Si le code bugue lorsque vous l'installez sur votre serveur, il faudra vous débrouiller
- ✓ Il faut que vous sachiez décrire les fonctionnalités de votre application, sinon, Lovable improvisera...
- ✓ Vous devrez sécuriser votre application.

Mais, c'est une révolution pour les non codeurs

Les coûts de développement divisés par 10 ?

- ✓ Les entrepreneurs peuvent créer un MVP, tous seuls
- ✓ Les gens des métiers peuvent créer des applications dont ils rêvent (ou presque) pour quelques milliers d'euros, même si l'on intègre, dans ce montant, l'intervention en bout de course d'un développeur pour installer et maintenir l'application
- ✓ Les start-ups peuvent lancer une version alpha pour valider leur concept
- ✓ Les petits malins peuvent aussi intégrer les APIs d'OpenAI ou de Mistral dans des applications simples, sans développeur.

Mais, c'est une révolution pour les non codeurs

Véritable révolution, trois mois après son lancement en novembre 2024, Lovable est devenu un phénomène de masse

- ✓ 25 000 applications créées chaque jour par 500 000 utilisateurs, dont 30000 abonnés payants.

Lovable

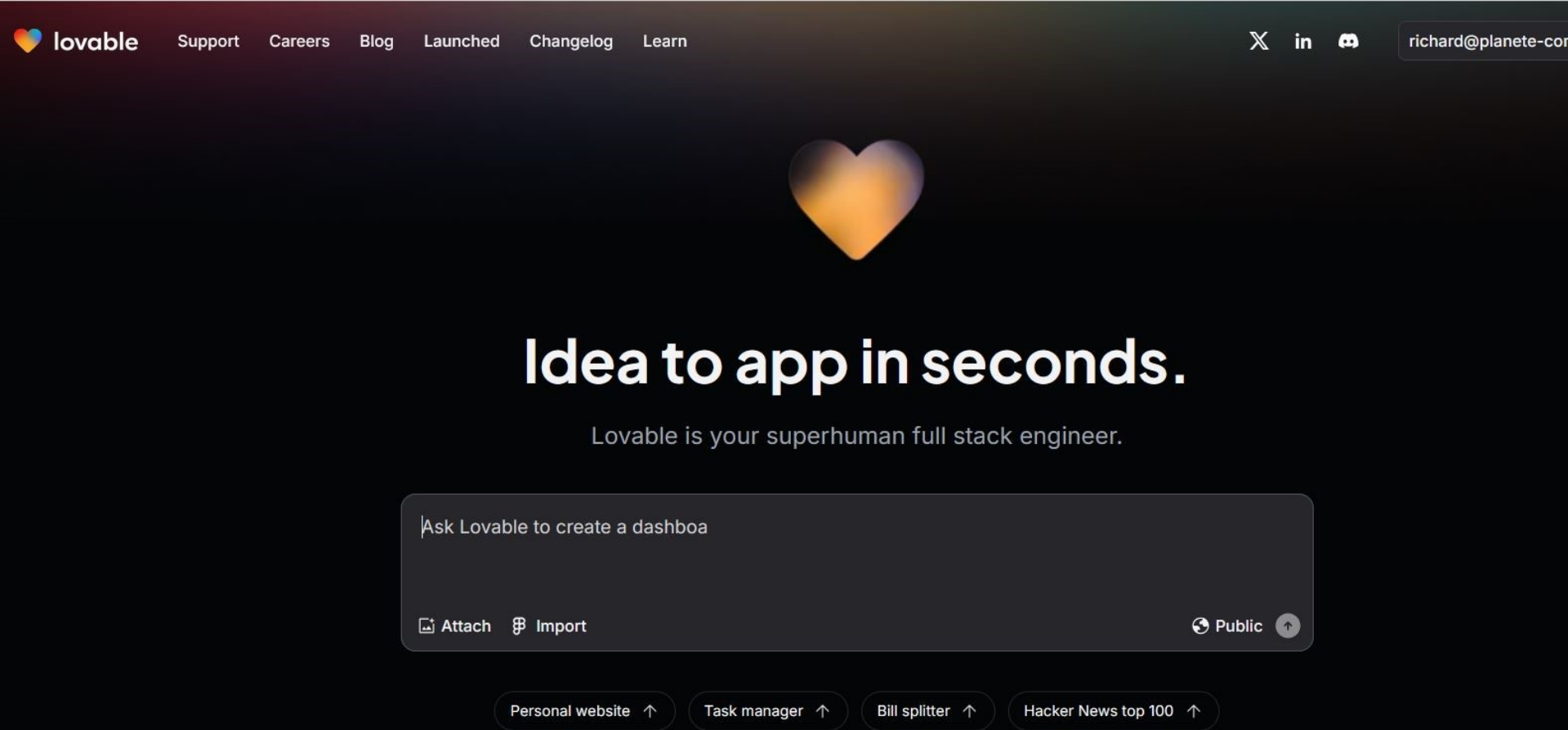
Par où commencer ?




Raphael.richard@neodia.com


06 62 55 00 10

La home page nous accueille avec une boîte de dialogue



lovable Support Careers Blog Launched Changelog Learn





X in  richard@planete-con



Idea to app in seconds.

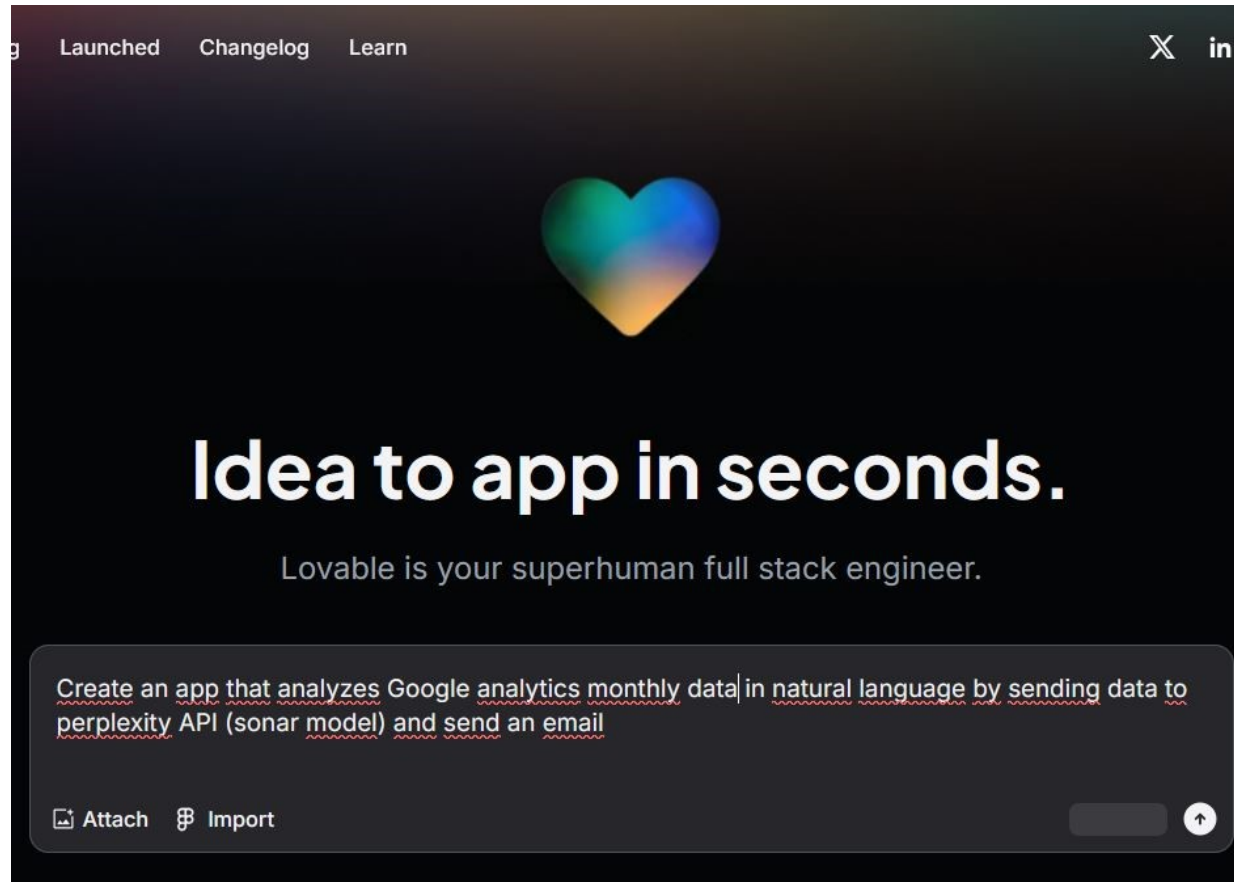
Lovable is your superhuman full stack engineer.

Ask Lovable to create a dashboa

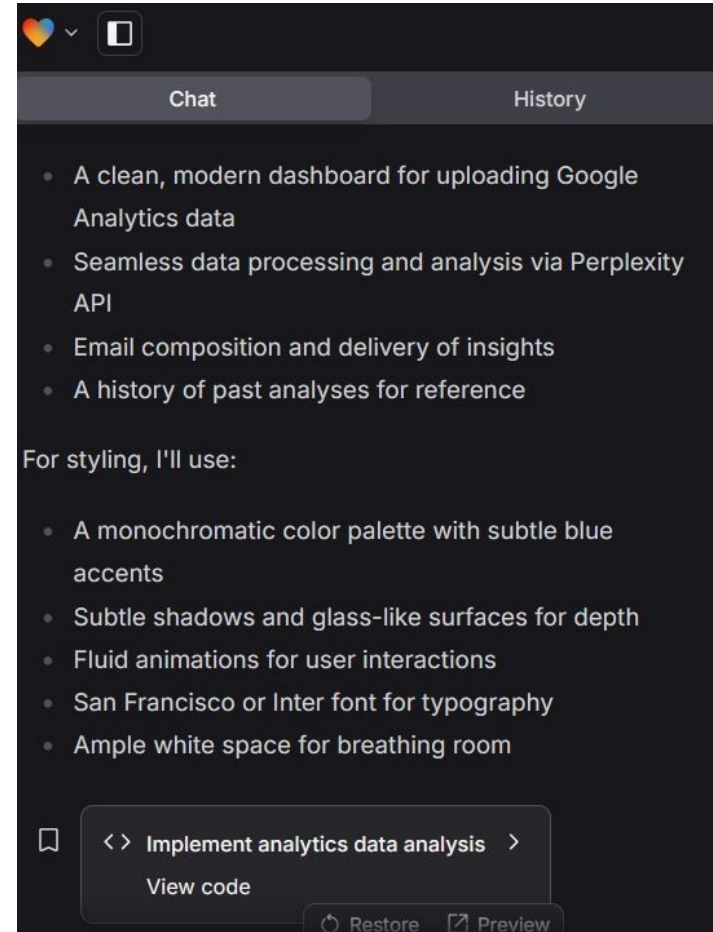
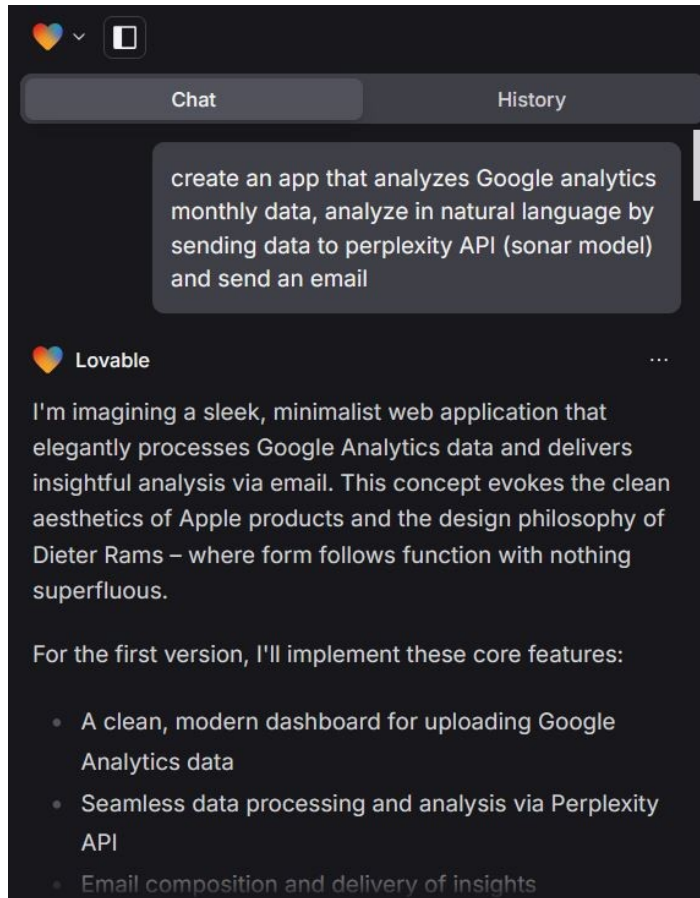
 Attach  Import  Public 

Personal website ↑ Task manager ↑ Bill splitter ↑ Hacker News top 100 ↑

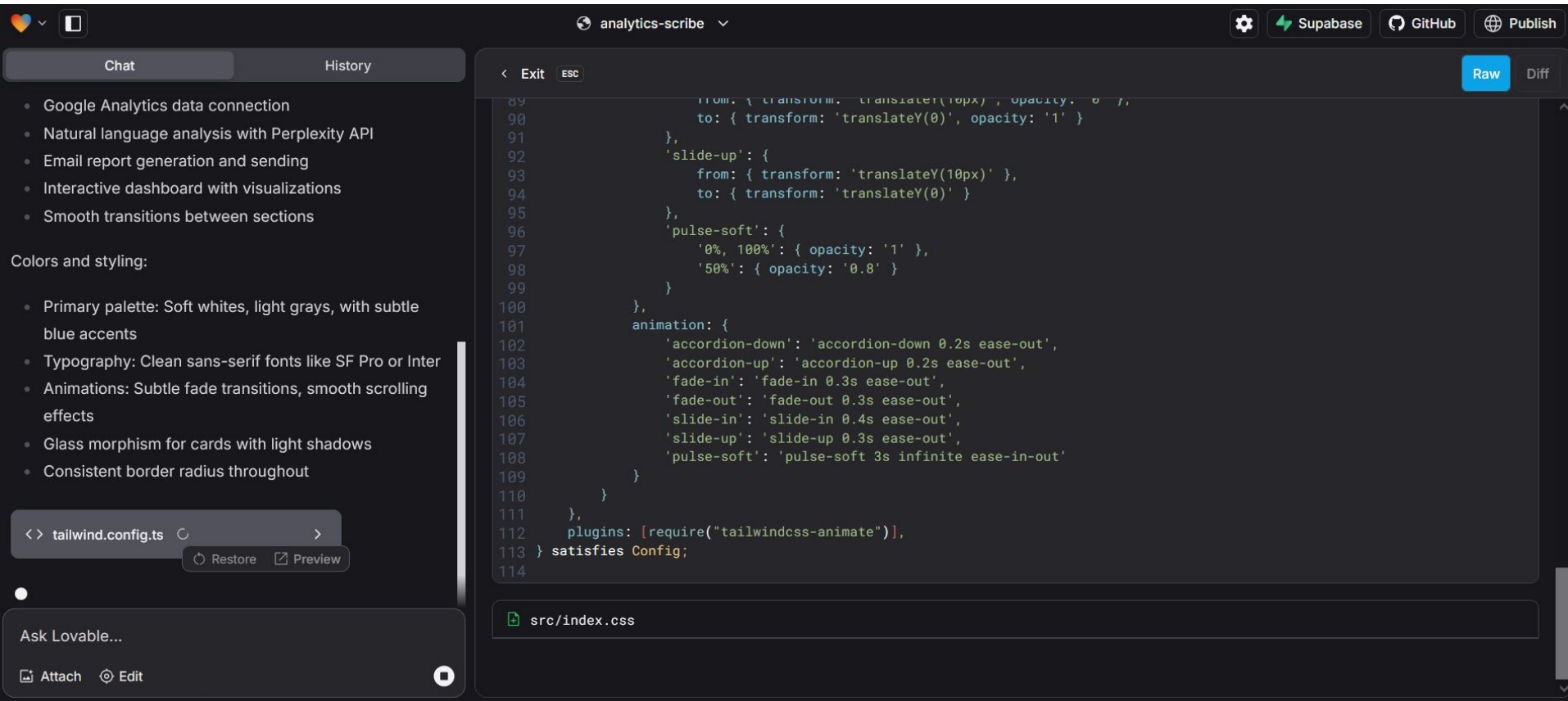
Pour un premier essai, décrivez de façon spécifique mais simple votre application



Lovable reformule et complète votre demande pour avoir le minimum d'info vital



Lovable commence alors à structurer l'application, lister les fichiers nécessaires et à générer le code



The screenshot displays the Lovable AI interface. On the left, a chat window titled "Chat" shows a list of requirements and styling instructions. On the right, a code editor displays the generated Tailwind CSS configuration for animations.

Chat Content:

- Google Analytics data connection
- Natural language analysis with Perplexity API
- Email report generation and sending
- Interactive dashboard with visualizations
- Smooth transitions between sections

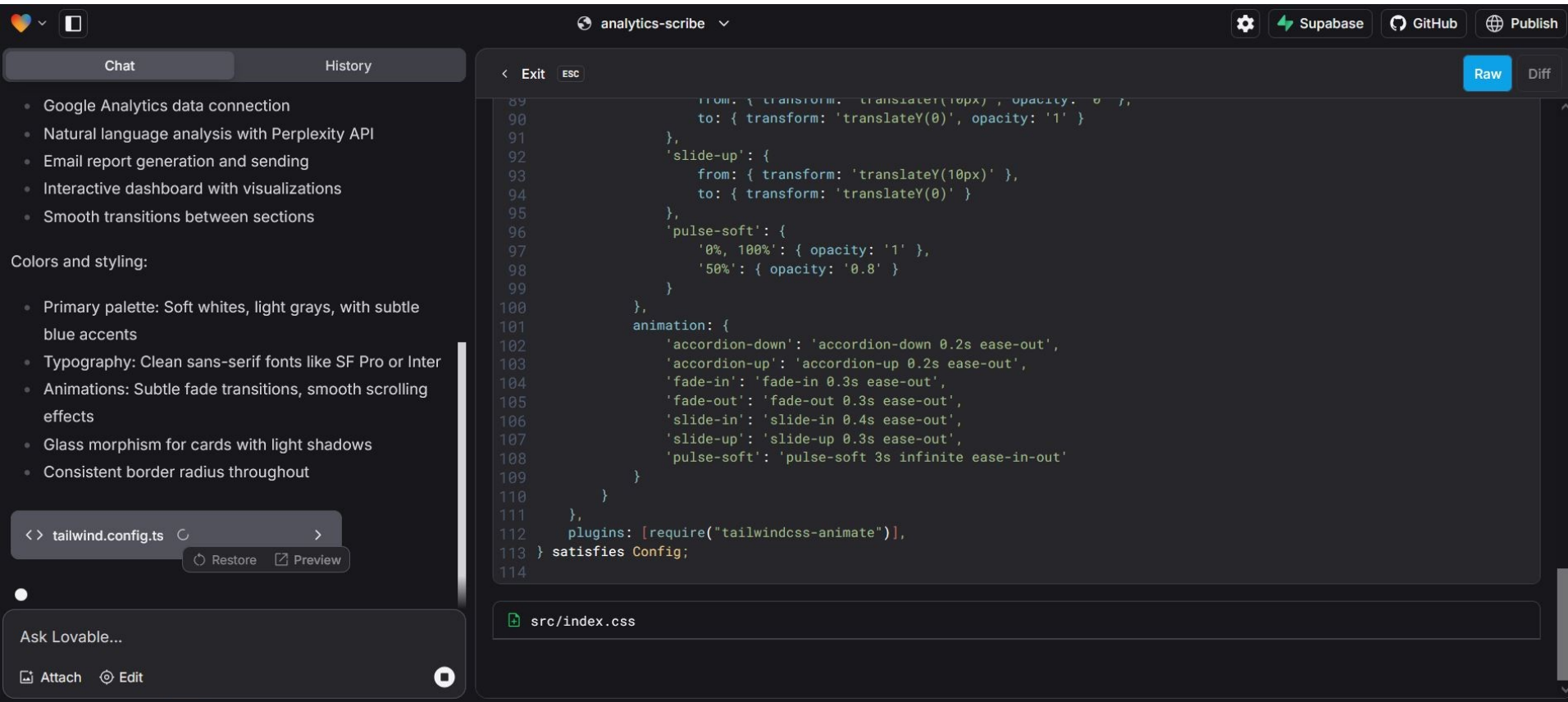
Colors and styling:

- Primary palette: Soft whites, light grays, with subtle blue accents
- Typography: Clean sans-serif fonts like SF Pro or Inter
- Animations: Subtle fade transitions, smooth scrolling effects
- Glass morphism for cards with light shadows
- Consistent border radius throughout

Code Editor Content (src/index.css):

```
109     from: { transform: 'translate(10px)', opacity: '0.8' },
110     to: { transform: 'translateY(0)', opacity: '1' }
111   },
112   'slide-up': {
113     from: { transform: 'translateY(10px)' },
114     to: { transform: 'translateY(0)' }
115   },
116   'pulse-soft': {
117     '0%, 100%': { opacity: '1' },
118     '50%': { opacity: '0.8' }
119   }
120 },
121 animation: {
122   'accordion-down': 'accordion-down 0.2s ease-out',
123   'accordion-up': 'accordion-up 0.2s ease-out',
124   'fade-in': 'fade-in 0.3s ease-out',
125   'fade-out': 'fade-out 0.3s ease-out',
126   'slide-in': 'slide-in 0.4s ease-out',
127   'slide-up': 'slide-up 0.3s ease-out',
128   'pulse-soft': 'pulse-soft 3s infinite ease-in-out'
129 }
130 },
131 },
132 plugins: [require("tailwindcss-animate")],
133 } satisfies Config;
134
```

Vous avez alors basculé sur la véritable interface de Lovable



The screenshot displays the Lovable AI interface. On the left, a chat window titled 'analytics-scribe' is open, showing a list of project features and styling instructions. The chat window includes a 'Chat' tab, a 'History' tab, and a 'Ask Lovable...' input field at the bottom. The main content area on the right is a code editor showing a CSS configuration file for Tailwind CSS. The code defines an animation object with various transitions and a plugin for Tailwind CSS animate.

Chat Content:

- Google Analytics data connection
- Natural language analysis with Perplexity API
- Email report generation and sending
- Interactive dashboard with visualizations
- Smooth transitions between sections

Colors and styling:

- Primary palette: Soft whites, light grays, with subtle blue accents
- Typography: Clean sans-serif fonts like SF Pro or Inter
- Animations: Subtle fade transitions, smooth scrolling effects
- Glass morphism for cards with light shadows
- Consistent border radius throughout

Code Editor Content (src/index.css):

```
109     from: { transform: 'translateY(10px)', opacity: '0' },
110     to: { transform: 'translateY(0)', opacity: '1' }
111   },
112   'slide-up': {
113     from: { transform: 'translateY(10px)' },
114     to: { transform: 'translateY(0)' }
115   },
116   'pulse-soft': {
117     '0%, 100%': { opacity: '1' },
118     '50%': { opacity: '0.8' }
119   }
120 },
121 animation: {
122   'accordion-down': 'accordion-down 0.2s ease-out',
123   'accordion-up': 'accordion-up 0.2s ease-out',
124   'fade-in': 'fade-in 0.3s ease-out',
125   'fade-out': 'fade-out 0.3s ease-out',
126   'slide-in': 'slide-in 0.4s ease-out',
127   'slide-up': 'slide-up 0.3s ease-out',
128   'pulse-soft': 'pulse-soft 3s infinite ease-in-out'
129 }
130 },
131 plugins: [require("tailwindcss-animate")],
132 } satisfies Config;
133
```

Cette interface ressemble à celle de Canvas de ChatGPT : le contenu créé par ChatGPT d'un côté et une colonne pour dialoguer et affiner l'application (à gauche)

La colonne de dialogue pour faire avancer le projet

La production du chatbot avec le code généré

The screenshot displays a web development IDE interface. On the left side, there is a chat window titled "Chat" with a "History" tab. The chat content includes a message: "I've created a sleek web application that allows users to upload Google Analytics data for AI analysis and email the insights." Below this, a section titled "What's next?" lists several tasks:

- **Refine & Customize:** Tweak the design, animations, and layouts via prompts or visual edits.
- **Master Prompting:** Use clear, detailed, and iterative prompts for better outcomes.
- **Expand with Backend:** Connect Supabase to add login, store data or advanced AI features.
- **GitHub Sync:** Transfer your project's code to GitHub for two-way sync of edits.
- **Debug with Ease:** Activate our experimental "chat mode" to troubleshoot issues quickly.
- **Add project knowledge:** Set key context or custom instructions you want to include in every edit in this project.

At the bottom of the chat window, there is an input field labeled "Ask Lovable..." and buttons for "Attach" and "Edit".

The right side of the IDE shows a code editor with the file "src/pages/Index.tsx" open. The code is as follows:

```
1
2 import React, { useState } from 'react';
3 import Header from "@/components/Header";
4 import UploadForm from "@/components/UploadForm";
5 import ResultsView from "@/components/ResultsView";
6 import EmailForm from "@/components/EmailForm";
7 import { analyzeWithPerplexity, hasApiKey, saveApiKey } from "@/utils/perplexity";
8 import { useToast } from "@/hooks/use-toast";
9 import { Card, CardContent, CardDescription, CardFooter, CardHeader, CardTitle } from "react-bootstrap";
10 import { Input } from "@/components/ui/input";
11 import { Button } from "@/components/ui/button";
12 import { Key, AlertCircle } from "lucide-react";
13 import { Alert, AlertDescription, AlertTitle } from "@/components/ui/alert";
14
15 const Index = () => {
16   const [analyticsData, setAnalyticsData] = useState<any>(null);
17   const [analysisResult, setAnalysisResult] = useState<string | null>(null);
18   const [isAnalyzing, setIsAnalyzing] = useState(false);
19   const [showEmailModal, setShowEmailModal] = useState(false);
20   const [apiKey, setApiKey] = useState("");
21   const [apiKeySet, setApiKeySet] = useState(hasApiKey());
22   const { toast } = useToast();
23
24   const handleUploadSuccess = async (data: any) => {
25     setAnalyticsData(data);
26
27     if (!apiKeySet) {
```

At the bottom of the code editor, there is a "Read Only. If you want to update the code, set up sync with Github." message.

15

Suivant la complexité de l'application, Lovable a besoin de 2 minutes à 15 minutes pour générer son code. Une fois qu'il a terminé, vous basculez sur le prototype de l'application

The screenshot displays a web development environment. On the left, a chat window is open, showing a list of issues and their resolutions. The issues include missing closing tags for a form element and a mismatch in closing tags structure. A solution is provided to fix these issues. Below the chat, a code editor snippet is visible, showing a fix for a circular dependency in CSS. The right side of the screen shows a preview of the application, titled 'Analytics Insight'. The application page has a header with the title and a 'Dashboard' link. The main content area features a large heading 'Analytics Insight' and a sub-heading 'Upload your Google Analytics data for AI-powered analysis and insights'. Below this, there is a section titled 'Perplexity API Key Required' with a warning icon and text stating 'To analyze your data, please provide your Perplexity API key'. A text input field is provided for the API key, and a blue 'Save Key' button is next to it. At the bottom, there is a section titled 'Upload Analytics Data' with a sub-heading 'Upload your Google Analytics monthly export file (JSON or CSV format)'.

Chat History

1. There are missing closing tags for the form element
2. There's a mismatch in closing tags structure

Let's fix both issues:

<> Fix circular dependency in CSS >

View code

Restore Preview

I fixed two issues:

1. The CSS circular dependency in index.css by replacing the `.transition-all` class with a new `.transition-custom` class that directly defines the transition properties instead of using `@apply transition-all`.
2. Fixed multiple syntax errors in the UploadForm.tsx component by correcting the form structure, properly closing all tags, and organizing the component's HTML structure.

Refactor UploadForm.tsx

Ask Lovable...

Attach Edit

analytics-sentiment-scribe

preview--analytics-sentiment-scribe.lovable.app / index

Analytics Insight

Dashboard

Analytics Insight

Upload your Google Analytics data for AI-powered analysis and insights

Perplexity API Key Required

To analyze your data, please provide your Perplexity API key

Important
Your API key is stored locally in your browser and is never sent to our servers.

Enter your Perplexity API key

Save Key

Upload Analytics Data

Upload your Google Analytics monthly export file (JSON or CSV format)

Dans le cadre de cette application qui doit analyser des données Google Analytics grâce à un modèle de Perplexity, Lovable a besoin de notre clé API Perplexity

The image shows a web application interface with a chat window on the left and a main content area on the right.

Chat Window:

- Buttons: Chat, History
- Issues:
 - There are missing closing tags for the form element
 - There's a mismatch in closing tags structure
- Message: "Let's fix both issues:"
- Code Fix Suggestion:

```
<> Fix circular dependency in CSS >
```

View code

Restore Preview
- Message: "I fixed two issues:"
- Fixes:
 - The CSS circular dependency in index.css by replacing the `.transition-all` class with a new `.transition-custom` class that directly defines the transition properties instead of using `@apply transition-all`.
 - Fixed multiple syntax errors in the UploadForm.tsx component by correcting the form structure, properly closing all tags, and organizing the component's HTML structure.
- Button: Refactor UploadForm.tsx
- Input: Ask Lovable...
- Buttons: Attach, Edit

Main Content Area:

- Page Title: analytics-sentiment-scribe
- Page URL: preview--analytics-sentiment-scribe.lovable.app / index
- Page Title: Analytics Insight
- Page Subtitle: Analytics Insight
- Text: Upload your Google Analytics data for AI-powered analysis and insights
- Section: Perplexity API Key Required
- Text: To analyze your data, please provide your Perplexity API key
- Important Message: Your API key is stored locally in your browser and is never sent to our servers.
- Input: Enter your Perplexity API key
- Button: Save Key
- Section: Upload Analytics Data
- Text: Upload your Google Analytics monthly export file (JSON or CSV format)

17

Nous récupérons notre clé API chez Perplexity (sur le compte que nous avons créé il y a un an et demi lorsque nous avons commencé à développer à l'ancienne des applis reposant sur l'API)

perplexity

⚠ Veuillez compléter le processus de paiement pour utiliser les fonctions de Enterprise Pro →

Régleres [Compte client](#) [Profil](#) [Personnaliser](#) [Connecteurs](#) [Fichiers](#) **API** [Organisation](#)

Nouveau fil d... [Ctrl](#) [I](#)

🔍 Accueil

🌐 Découvrir

🏠 Espaces

📖 Bibliothèque

quels sont les LLMs ou ch

J'ai installé FS poster sur

peux tu me présenter l'off

📉 Utilisation actuelle

\$0.06

📈 Niveau d'utilisation API

Niveau 1


La clé API

🔑 Clés API [+ Générer](#)

pp1: [redacted] 8f05ccd0a [📄](#) [🗑](#)

👤 raphaelric [⚙](#)

Nous collons notre clé API récupérée chez Perplexity dans le champ prévu à cet effet dans l'application créée par Lovable

 Analytics Insight

[Dashboard](#)

Analytics Insight

Upload your Google Analytics data for AI-powered analysis and insights

Perplexity API Key Required

To analyze your data, please provide your Perplexity API key.

 **Important**

Your API key is stored locally in your browser and is never sent to our servers.

La clé API

.....

 Save Key

Upload Analytics Data

Upload your Google Analytics monthly export file (.JSON or CSV format)

Nous allons pouvoir tester si notre application fonctionne effectivement

The image shows a development environment with a chat window on the left and a preview of an application on the right.

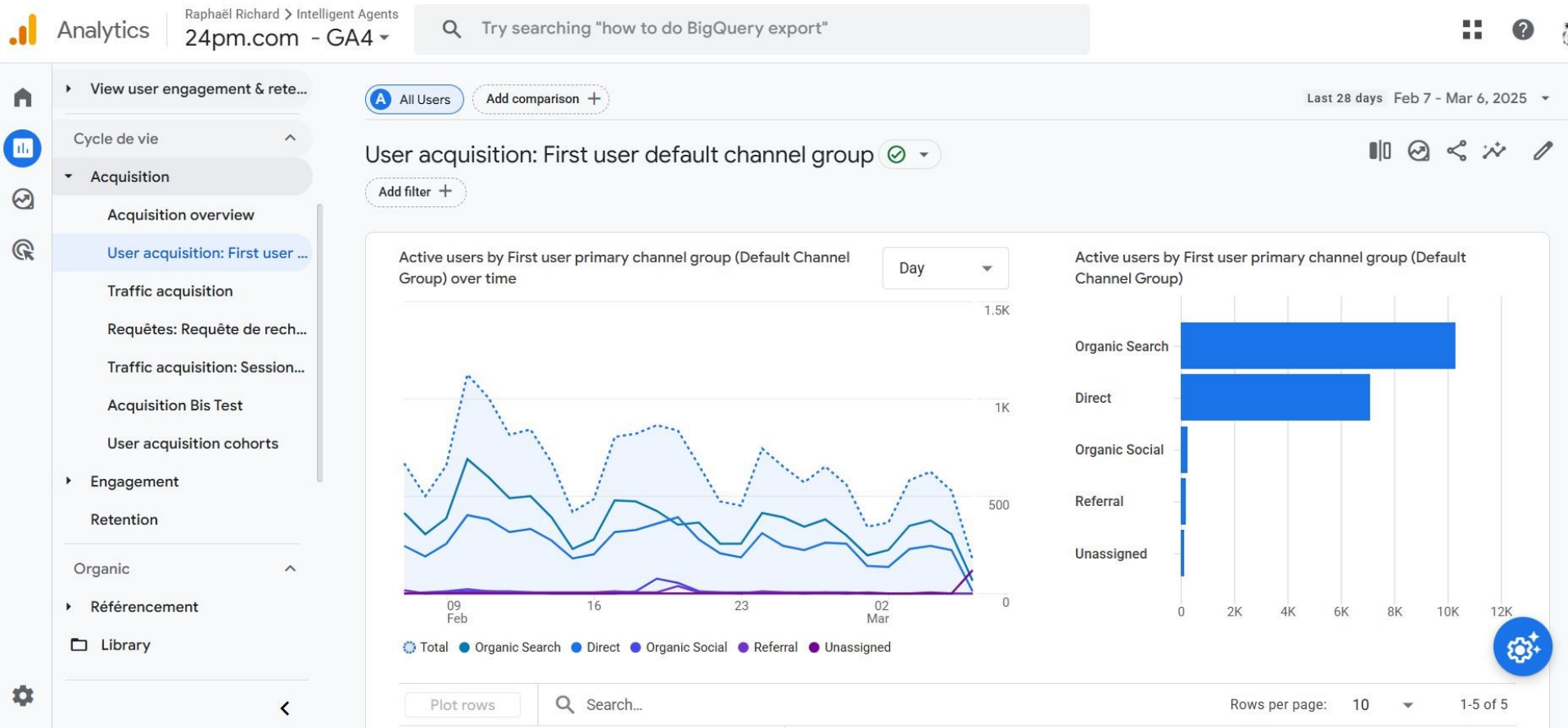
Chat Window (Left):

- Buttons: Chat, History
- Issues:
 1. There are missing closing tags for the form element
 2. There's a mismatch in closing tags structure
- Message: "Let's fix both issues:"
- Code Snippet: `<> Fix circular dependency in CSS >`
- Buttons: View code, Restore, Preview
- Message: "I fixed two issues:"
- Issues:
 1. The CSS circular dependency in `index.css` by replacing the `.transition-all` class with a new `.transition-custom` class that directly defines the transition properties instead of using `@apply transition-all`.
 2. Fixed multiple syntax errors in the `UploadForm.tsx` component by correcting the form structure, properly closing all tags, and organizing the component's HTML structure.
- Buttons: Refactor UploadForm.tsx
- Input: Ask Lovable...
- Buttons: Attach, Edit

Preview Window (Right):

- URL: `preview--analytics-sentiment-scribe.lovable.app / index`
- Page Title: Analytics Insight
- Page Content:
 - Analytics Insight
 - Dashboard
 - Upload your Google Analytics data for AI-powered analysis and insights
 - Upload Analytics Data
 - Upload your Google Analytics monthly export file (JSON or CSV format)
 - Drag and drop your file here or click to browse
 - Maximum file size: 5MB

Pour ce faire, nous nous affichons sur un rapport sur Google Analytics 4



21

Nous cliquons sur le picto qui nous permet de basculer sur l'explorateur de données de GA4

Explorateur



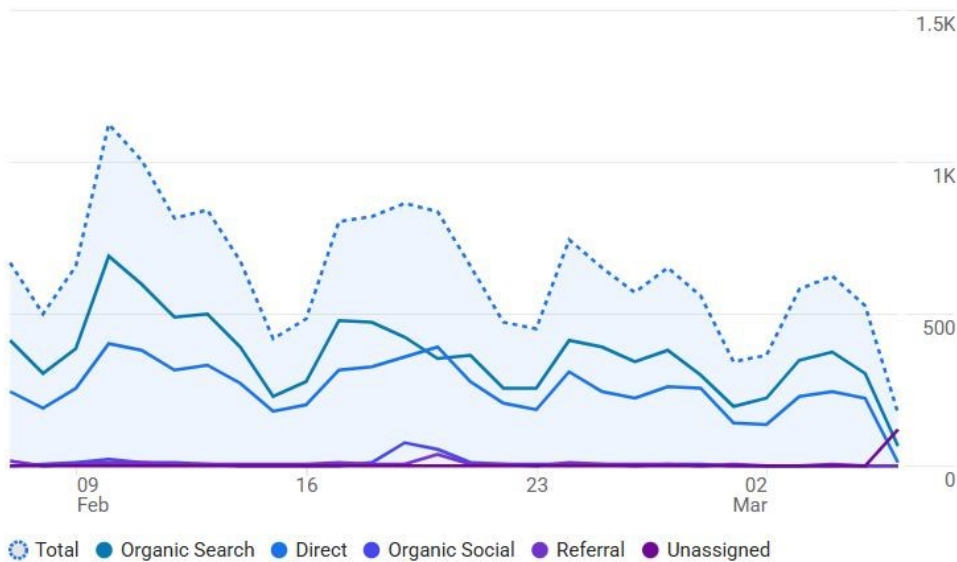
User acquisition: First user default channel group



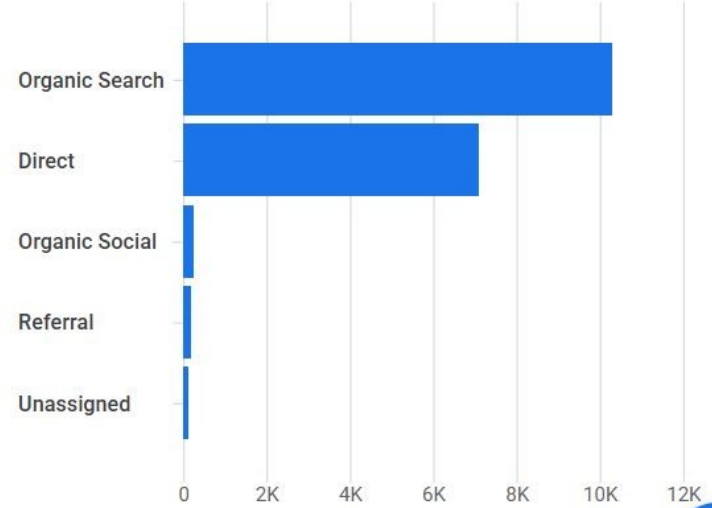
Open this report as an exploration

Add filter +

Active users by First user primary channel group (Default Channel Group) over time



Active users by First user primary channel group (Default Channel Group)



Nous arrivons dans l'explorateur

Analytics | 24pm.com - GA4 | 24pm.com - GA4

Try searching "how to do BigQuery export"

Variables X Settings X

EXPLORATION NAME:
User acquisition: First user...

Custom
Feb 7 - Mar 6, 2025

SEGMENTS +

DIMENSIONS +

First user primary channel group (Default Channel Group)

METRICS +

- Active users
- New users
- Engaged sessions
- Engagement rate

TECHNIQUE

Free form

VISUALIZATION

SEGMENT COMPARISONS

+ Drop or select segment

ROWS

First user primary channel group (Default Channel Group)

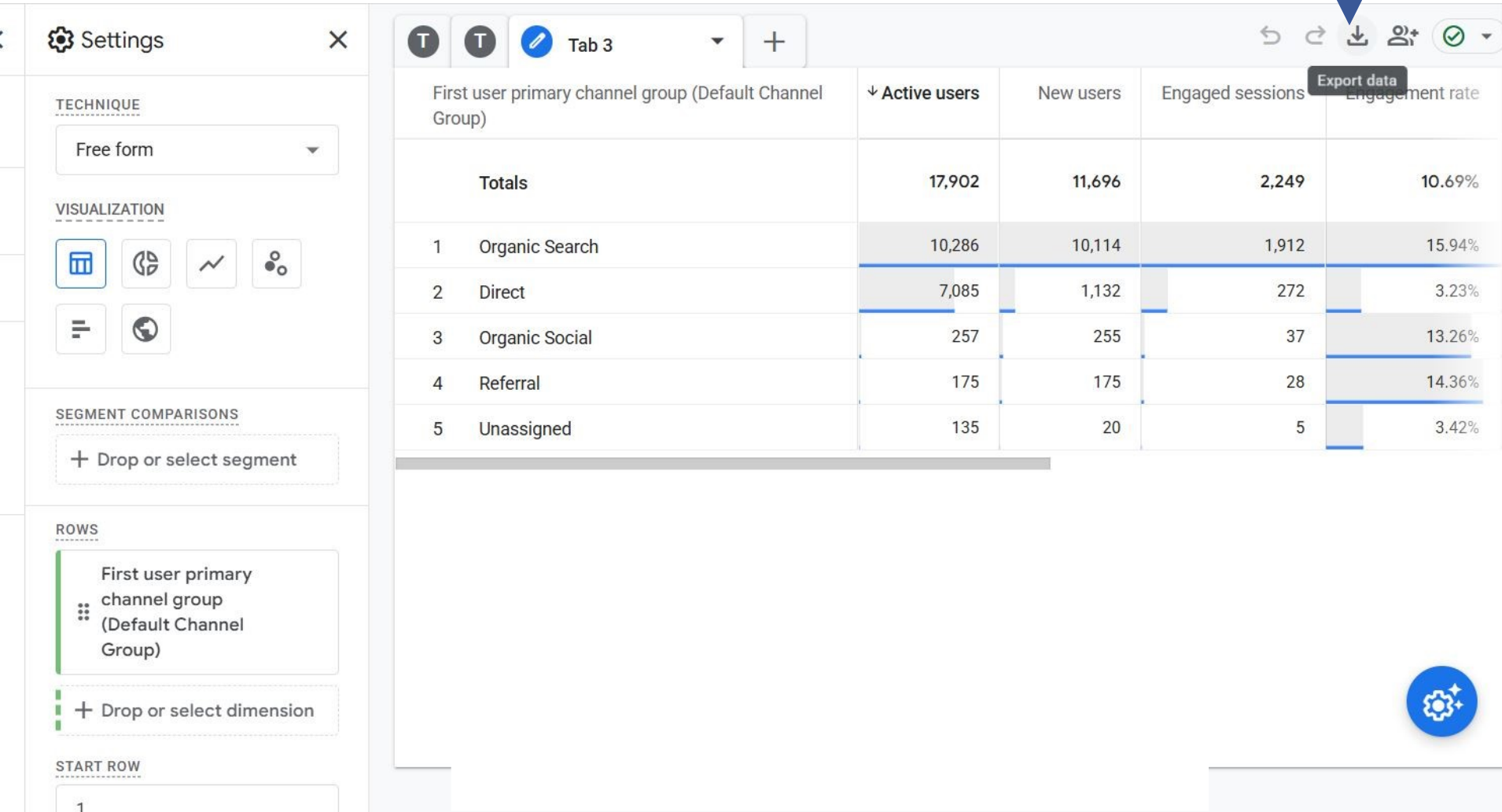
+ Drop or select dimension

START ROW

First user primary channel group (Default Channel Group)		↓ Active users	New users	Engaged sessions	Engagement rate
Totals		17,902	11,696	2,249	10.69%
1	Organic Search	10,286	10,114	1,912	15.94%
2	Direct	7,085	1,132	272	3.23%
3	Organic Social	257	255	37	13.26%
4	Referral	175	175	28	14.36%
5	Unassigned	135	20	5	3.42%

Nous exportons les données au format CSV

Exporter les données
en CSV



The screenshot displays a data analytics dashboard with a table of user engagement metrics. The table is titled 'First user primary channel group (Default Channel Group)' and includes columns for 'Active users', 'New users', 'Engaged sessions', and 'Engagement rate'. The data is organized into a 'Totals' row and five numbered rows representing different channel groups: Organic Search, Direct, Organic Social, Referral, and Unassigned. A blue callout box with the text 'Exporter les données en CSV' points to the 'Export data' button in the top right corner of the table.

	First user primary channel group (Default Channel Group)	Active users	New users	Engaged sessions	Engagement rate
Totals		17,902	11,696	2,249	10.69%
1	Organic Search	10,286	10,114	1,912	15.94%
2	Direct	7,085	1,132	272	3.23%
3	Organic Social	257	255	37	13.26%
4	Referral	175	175	28	14.36%
5	Unassigned	135	20	5	3.42%

Nous faisons glisser le fichier CSV dans notre application
(en mode test, l'option « browse » ne fonctionnait)

The screenshot shows a web browser window displaying a web application. The browser's address bar shows the URL `preview--analytics-sentiment-scribe.lovable.app / index`. The application interface includes a header with the text `analytics-sentiment-scribe` and navigation links for `Dashboard` and `History`. The main content area features the heading `Upload Analytics Data` and the instruction `Upload your Google Analytics monthly export file (JSON or CSV format)`. A large dashed box contains the text `Drag and drop your file here` and `or click to browse`, with an upload icon above it. Below this, a blue button labeled `Analyze Data` is visible. A file explorer window is overlaid on the application, showing the `Téléchargements` folder. A blue box with the text `Faire glisser le fichier CSV` is positioned over the file explorer, with an arrow pointing to the `download (3).csv` file. Another arrow points from this file to the upload area in the application. The browser's developer console shows a message: `1. There are missing closing tags for the form element`. The bottom of the image features the `24pm` academy logo.

Ensuite, notre application envoie les données à l'API de Perplexity et récupère l'analyse.

The image shows a screenshot of a web application interface. On the left, there is a dark sidebar with a 'Chat' window. The chat window has a 'History' tab and shows two messages from 'Yesterday':

- Message 1: "Fix circular dependency in CSS" (6 mars 2025, 18:06)
- Message 2: "Implement analytics data analysis" (6 mars 2025, 18:05)

At the bottom of the chat window, there is a 'Previewing' section with an 'Exit' button and a 'Restore this version' button.

The main content area is a web browser window showing a dashboard for 'Analytics Insight'. The browser address bar displays 'preview--analytics-sentiment-scribe.lovable.app / index'. The dashboard has a 'Dashboard' link in the top right corner. The main heading is 'Analysis Results', followed by a sub-heading 'Monthly Analytics Report Summary - 07/03/2025'. The content includes:

- Users and Sessions Overview:** Your website experienced a 12% increase in total users this month compared to last month, with 43,250 unique visitors. Session duration averaged 2:45 minutes, representing a slight improvement of 8% from the previous period. The bounce rate decreased from 58% to 52%, indicating better user engagement with your content.
- Traffic Sources Analysis:** Organic search continues to be your primary traffic driver at 42%, though social media referrals showed the most significant growth this month (+23%). Direct traffic accounts for 28% of sessions, while paid campaigns contributed 18% with a conversion rate of 3.2%, outperforming other channels. Mobile traffic now represents 68% of all sessions, emphasizing the importance of mobile optimization.
- Conversion Funnel Performance:** The overall conversion rate improved to 2.8% (up from 2.3% last month). Product page

Puis, l'application me propose d'envoyer le rapport par email

The screenshot displays a web application interface. On the left, there is a dark sidebar with a 'Chat' window showing two messages from 'Yesterday' dated '6 mars 2025, 18:06' and '6 mars 2025, 18:05'. The main content area is titled 'Analytics Insight' and contains a report. The report text states: 'The overall conversion rate improved to 2.8% (up from 2.3% last month). Product page views increased by 15%, but cart abandonment remains high at 72%. The checkout completion rate showed modest improvement at 48% (+5% from last period). The most significant drop-offs occur at the payment information stage, suggesting potential friction points to address.' Below this, 'Recommendations:' are listed: 1. Optimize the payment stage of your checkout process. 2. Allocate more resources to your social media marketing. 3. Consider creating more mobile-specific content given the high percentage of mobile users. 4. Investigate which organic search terms are performing best and create more content around those topics. At the bottom right of the report, a blue button labeled 'Send via Email' is highlighted with a blue box and an arrow pointing to it. Other buttons include 'Copy to Clipboard' and 'Dashboard'.

analytics-sentiment-scribe

preview--analytics-sentiment-scribe.lovable.app / index

Analytics Insight

Dashboard History

The overall conversion rate improved to 2.8% (up from 2.3% last month). Product page views increased by 15%, but cart abandonment remains high at 72%. The checkout completion rate showed modest improvement at 48% (+5% from last period). The most significant drop-offs occur at the payment information stage, suggesting potential friction points to address.

Recommendations:

1. Optimize the payment stage of your checkout process.
2. Allocate more resources to your social media marketing.
3. Consider creating more mobile-specific content given the high percentage of mobile users.
4. Investigate which organic search terms are performing best and create more content around those topics.

Copy to Clipboard

Send via Email

Envoyer le rapport par email

Mais, il y a un problème...

Je voulais une application qui récupère automatiquement les données de GA toutes les semaines et envoie automatiquement l'analyse de Perplexity par email.

Pourquoi Lovable n'a pas fait cela ?

Simplement, parce que je ne lui ai pas demandé clairement !
(vérifiez le sur la page où j'ai décrit mon application).

Avec Lovable comme avec un développeur, il faut tout préciser.

Mais bonne nouvelle, cela peut se faire dans le prompt initial, mais aussi ensuite via la colonne de dialogue à gauche, une fois la première version générée.

Pour finaliser cette application, il faut encore

- Que j'explique à Lovable que mon application doit récupérer les données de Google Analytics toutes les semaines

Il me demandera alors ma clé API Google Analytics ainsi que la liste des rapports qu'il doit récupérer

- Que je donne à Lovable mes propres règles d'analyse des données (sinon Perplexity analysera les données comme il pourra)
- Que je donne des consignes pour finaliser le design (le style général que j'attends, mais aussi certains éléments précis à ajouter)
- Que je demande à Lovable les éléments manquants (CTA, pages, textes...)

Imaginons que l'application soit finalisée et que je désire la passer en production...

J'ai alors deux choix:

1. La « publier » sur Lovable qui l'hébergera (je pourrai alors faire pointer l'un de mes noms de domaine sur cette application)
2. La télécharger pour l'installer sur l'un de mes serveurs. Il faudra alors que je demande à Lovable de se connecter à mon compte Github pour qu'il y transfère l'application. De là, je pourrais basculer mon application sur mon serveur/hébergement mutualisé avec tous les avantages que procure Github;

Explorons maintenant l'interface de Lovable

The image shows a split-screen view of a web development environment. On the left, a dark-themed chat window is open, displaying a list of issues and their resolutions. The issues include missing closing tags for a form element and a mismatch in closing tags structure. The chat also shows a fix for a circular dependency in CSS and multiple syntax errors in the UploadForm.tsx component. At the bottom of the chat, there is an 'Ask' input field with a 'Select to edit (Alt-S)' button, and 'Attach' and 'Edit' buttons.

On the right, a preview of a web page titled 'Analytics Insight' is shown. The page has a header with the title and a 'Dashboard' link. The main content area features a large heading 'Analytics Insight' and a sub-heading 'Upload your Google Analytics data for AI-powered analysis and insights'. Below this is a section titled 'Upload Analytics Data' with a sub-heading 'Upload your Google Analytics monthly export file (JSON or CSV format)'. A large dashed box contains an upload icon and the text 'Drag and drop your file here or click to browse'. At the bottom of this section, it says 'Maximum file size: 5MB'.

Pour modifier un élément de l'application, cliquez sur le bouton « Edit »

Chat History

1. There are missing closing tags for the form element
2. There's a mismatch in closing tags structure

Let's fix both issues:

<> Fix circular dependency in CSS >

View code

Restore Preview

I fixed two issues:

1. The CSS circular dependency in index.css by replacing the `.transition-all` class with a new `.transition-custom` class that directly defines the transition properties instead of using `@apply transition-all`.
2. Fixed multiple syntax errors in the UploadForm.tsx component by correcting the form structure, properly closing the form element's HTML

Le bouton Edit

Refactor UploadForm.tsx

Ask Select to edit (Alt-S)

Attach Edit

preview--analytics-sentiment-scribe.lovable.app / index

Analytics Insight

Dashboard

Analytics Insight

Upload your Google Analytics data for AI-powered analysis and insights

Upload Analytics Data

Upload your Google Analytics monthly export file (JSON or CSV format)

Drag and drop your file here
or click to browse

Maximum file size: 5MB

Sélectionnez alors l'élément à modifier

Chat History

View code

Restore Preview

I fixed two issues:

1. The CSS circular dependency in index.css by replacing the `.transition-all` class with a new `.transition-custom` class that directly defines the transition properties instead of using `@apply transition-all`.
2. Fixed multiple syntax errors in the UploadForm.tsx component by correcting the form structure, properly closing all tags, and organizing the component's HTML structure.

Refactor UploadForm.tsx

Edit

Select to edit or ask Lovable to modify

Discard Save

Ask Lovable...

Attach Edit

preview--

Analytics Insight

Dashboard

Modifions le titre

h1

Analytics Insight

Upload your Google Analytics data for AI-powered analysis and insights

Upload Analytics Data

Upload your Google Analytics monthly export file (JSON or CSV format)

Drag and drop your file here
or click to browse

Maximum file size: 5MB

Dans la colonne de dialogue apparaissent alors les paramètres modifiables

The image shows a web development tool interface. On the left, a chat window displays a message: "I fixed two issues: 1. The CSS circle the .transition custom class properties instead of using @apply transition-all." Below the chat is a style editor for the "main" element, with fields for Margin, Padding, Background, and Border radius. A blue callout box with the text "Les paramètres modifiables pour l'éléments" points to the style editor. On the right, a preview window shows a page titled "Analytics Insight" with a large heading and a section for "Upload Analytics Data". The preview window includes a "Dashboard" link and a "History" link. The top of the preview window shows the URL "preview--analytics-sentiment-scribe.lovable.app / index" and navigation icons for Supabase, GitHub, and a globe.

Chat

History

View code

Restore Preview

I fixed two issues:

1. The CSS circle the .transition custom class properties instead of using @apply transition-all .

main

Margin

Padding

Background

Border radius

Advanced

Discard Save

Ask Lovable to modify Main...

Attach Edit

analytics-sentiment-scribe

preview--analytics-sentiment-scribe.lovable.app / index

Analytics Insight

Dashboard History

h1

Analytics Insight

Upload your Google Analytics data for AI-powered analysis and insights

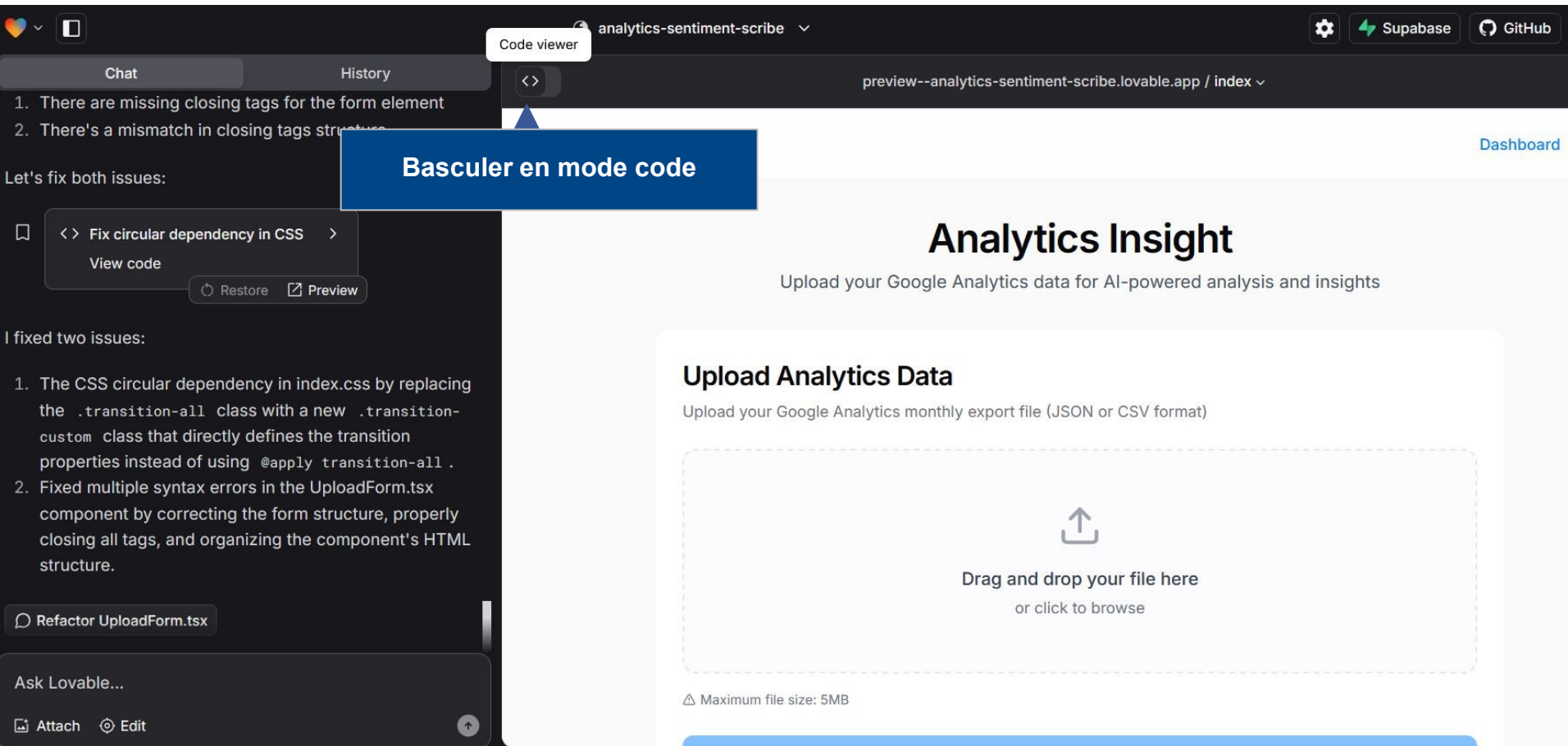
Upload Analytics Data

Upload your Google Analytics monthly export file (JSON or CSV format)

Drag and drop your file here
or click to browse

Maximum file size: 5MB

Vous pouvez basculer sur la vue « Code Viewer » en cliquant sur le picto `<>`



The screenshot shows a web application interface with a dark theme. The main content area displays "Analytics Insight" with a subheading "Upload your Google Analytics data for AI-powered analysis and insights". Below this is a section titled "Upload Analytics Data" with a subheading "Upload your Google Analytics monthly export file (JSON or CSV format)". A large dashed box contains an upload icon and the text "Drag and drop your file here or click to browse". At the bottom of this section, it says "Maximum file size: 5MB".

On the left side, there is a chat window with a list of issues and a "Code viewer" overlay. The overlay is a blue box with the text "Basculer en mode code" and a code icon `<>`. The chat window shows a list of issues and a "Fix circular dependency in CSS" button with a "View code" link. Below the chat window, there is a "Refactor UploadForm.tsx" button and an "Ask Lovable..." input field.

The top of the interface shows a navigation bar with a "Code viewer" dropdown menu, a "preview--analytics-sentiment-scribe.lovable.app / index" URL, and icons for "Supabase" and "GitHub".

Puis, vous pouvez revenir sur la vue « Application » en cliquant sur le même picto

The screenshot shows a code editor interface for a project named "analytics-sentiment-scribe". The interface includes a chat window on the left, a file explorer in the center, and a code editor on the right.

Chat History:

- 1. There are missing closing tags for the form element
- 2. There's a mismatch in closing tags structure

Let's fix both issues:

- <> Fix circular dependency in CSS >
- View code
- Restore
- Preview

I fixed two issues:

1. The CSS circular dependency in index.css by replacing the `.transition-all` class with a new `.transition-custom` class that directly defines the transition properties instead of using `@apply transition-all`.
2. Fixed multiple syntax errors in the UploadForm.tsx component by correcting the form structure, properly closing all tags, and organizing the component's HTML structure.

File Explorer:

- placemorder.svg
- src
 - components
 - hooks
 - lib
 - pages
 - utils
 - App.css
 - App.tsx
 - index.css
 - main.tsx
- components.json
- eslint.config.js
- index.html
- package.json
- postcss.config.js

Code Editor:

```

1
2 import React, { useState } from 'react';
3 import Header from "@components/Header";
4 import UploadForm from "@components/UploadForm";
5 import ResultsView from "@components/ResultsView";
6 import EmailForm from "@components/EmailForm";
7 import { analyzeWithPerplexity, hasApiKey, saveApiKey } from "@utils/perplexity";
8 import { useToast } from "@hooks/use-toast";
9 import { Card, CardContent, CardDescription, CardFooter, CardHeader, CardTitle } from "@components/ui/card";
10 import { Input } from "@components/ui/input";
11 import { Button } from "@components/ui/button";
12 import { Key, AlertCircle } from "lucide-react";
13 import { Alert, AlertDescription, AlertTitle } from "@components/ui/alert";
14
15 const Index = () => {
16   const [analyticsData, setAnalyticsData] = useState<any>(null);
17   const [analysisResult, setAnalysisResult] = useState<string | null>(null);
18   const [isAnalyzing, setIsAnalyzing] = useState(false);
19   const [showEmailModal, setShowEmailModal] = useState(false);
20   const [apiKey, setApiKey] = useState("");
21   const [apiKeySet, setApiKeySet] = useState(hasApiKey());
22   const { toast } = useToast();
23
24   const handleUploadSuccess = async (data: any) => {
25     setAnalyticsData(data);
26
27     if (!apiKeySet) {

```

Read Only. If you want to update the code, set up sync with Github.

L'onglet « History » vous permet de revenir sur les grandes étapes de la modification

The screenshot displays a code editor interface for a project named "analytics-sentiment-scribe". The top navigation bar includes "Chat", "History", and "Preview" tabs. The "History" tab is active, showing a list of recent changes under the heading "Yesterday".

Two blue callout boxes highlight key features:

- Onglet « history »**: Points to the "History" tab in the top navigation bar.
- Grandes étapes des modifications**: Points to the list of recent changes in the History tab.

The "History" tab shows the following changes:

- Fix circular dependency in CSS (6 mars 2025, 18:06)
- Implement analytics data analysis (6 mars 2025, 18:05)

The main editor area shows the code for "src/pages/Index.tsx". The code includes imports for React, Header, UploadForm, ResultsView, EmailForm, and various utility functions. The main component is defined as:

```
15 const Index = () => {
16   const [analyticsData, setAnalyticsData] = useState<any>(null);
17   const [analysisResult, setAnalysisResult] = useState<string | null>(null);
18   const [isAnalyzing, setIsAnalyzing] = useState(false);
19   const [showEmailModal, setShowEmailModal] = useState(false);
20   const [apiKey, setApiKey] = useState("");
21   const [apiKeySet, setApiKeySet] = useState(hasApiKey());
22   const { toast } = useToast();
23
24   const handleUploadSuccess = async (data: any) => {
25     setAnalyticsData(data);
26
27     if (!apiKeySet) {
```

Pour publier l'application et la faire héberger sur Lovable, cliquez sur « Publish »

Publier

The screenshot shows the Lovable IDE interface for a project named "analytics-sentiment-scribe". At the top right, there are buttons for "Supabase", "GitHub", and "Publish". The "Publish" button is highlighted with a blue arrow pointing to it from the text above. The main view displays the "Analytics Insight" application, which has a header "Analytics Insight" and a sub-header "Upload your Google Analytics data for AI-powered analysis and insights". Below this is a section titled "Upload Analytics Data" with the instruction "Upload your Google Analytics monthly export file (JSON or CSV format)". A dashed box contains an upload icon and the text "Drag and drop your file here or click to browse". Below the dashed box, it says "Maximum file size: 5MB".

On the left side, there is a chat window with the following content:

- Chat History
- 1. There are missing closing tags for the form element
- 2. There's a mismatch in closing tags structure
- Let's fix both issues:
- <> Fix circular dependency in CSS >
- View code
- Restore Preview
- fixed two issues:
- 1. The CSS circular dependency in index.css by replacing the .transition-all class with a new .transition-custom class that directly defines the transition properties instead of using @apply transition-all .
- 2. Fixed multiple syntax errors in the UploadForm.tsx component by correcting the form structure, properly closing all tags, and organizing the component's HTML structure.
- Refactor UploadForm.tsx
- Ask Lovable...
- Attach Edit

On the right side, there is a panel with the following content:

Publish your project to the following domains:

analytics-sentiment-scribe.lovable.app

Publish

Pour publier l'application et la faire héberger sur Lovable, cliquez sur « Publish »

The screenshot displays the Lovable interface for publishing an application. In the top right corner, there is a dark grey panel with the text "Publish your project to the following domains:" followed by the domain "analytics-sentiment-scribe.lovable.app" and a blue "Publish" button. A blue callout box with the text "L'application est live en 2 minutes" has a blue arrow pointing to a dark grey modal window. The modal window contains the text "Your site is live!" with a close button (X) in the top right corner. Below this, there is a white button labeled "Share on Lovable Launched" and a link "Want a custom domain? Learn more". The background of the interface shows the "Analytics Insight" header and a large "Analytics Insight" title. At the bottom, there is a dark grey area with an upload icon and the text "Drag and drop your file here or click to browse".

Pour publier l'application et la faire héberger sur Lovable, cliquez sur « Publish »

L'application est accessible sur un sous-domaine de Lovabl.app, mais vous pouvez faire pointer votre nom de domaine dessus

https://analytics-sentiment-scribe.lovable.app

Malaisie WEB AudioVidéo BtoB Automation FORMATION Msg Linkd Neodia-Gestion stats HH Perso Ecolo Google public Chaine Christine GA GTM

Analytics Insight Dashboard Hi

Analytics Insight

Upload your Google Analytics data for AI-powered analysis and insights

Perplexity API Key Required

To analyze your data, please provide your Perplexity API key

Important
Your API key is stored locally in your browser and is never sent to our servers.

Enter your Perplexity API key [Save Key](#)

Upload Analytics Data

Upload your Google Analytics monthly export file (JSON or CSV format)

Pour récupérer le code, connectez votre Lovable à votre compte Github



Vous ne pouvez pas télécharger le code sans passer par Github

Cliquez sur Github



The screenshot shows the Lovable IDE interface. On the left, a chat window displays a list of issues and a code fix. The chat messages are:

- 1. There are missing closing tags for the form element
- 2. There's a mismatch in closing tags structure

Let's fix both issues:

```
<> Fix circular dependency in CSS >
```

View code

Restore Preview

I fixed two issues:

- 1. The CSS circular dependency in index.css by replacing the `.transition-all` class with a new `.transition-custom` class that directly defines the transition properties instead of using `@apply transition-all`.
- 2. Fixed multiple syntax errors in the UploadForm.tsx component by correcting the form structure, properly closing all tags, and organizing the component's HTML structure.

Refactor UploadForm.tsx

Ask Lovable...

Attach Edit

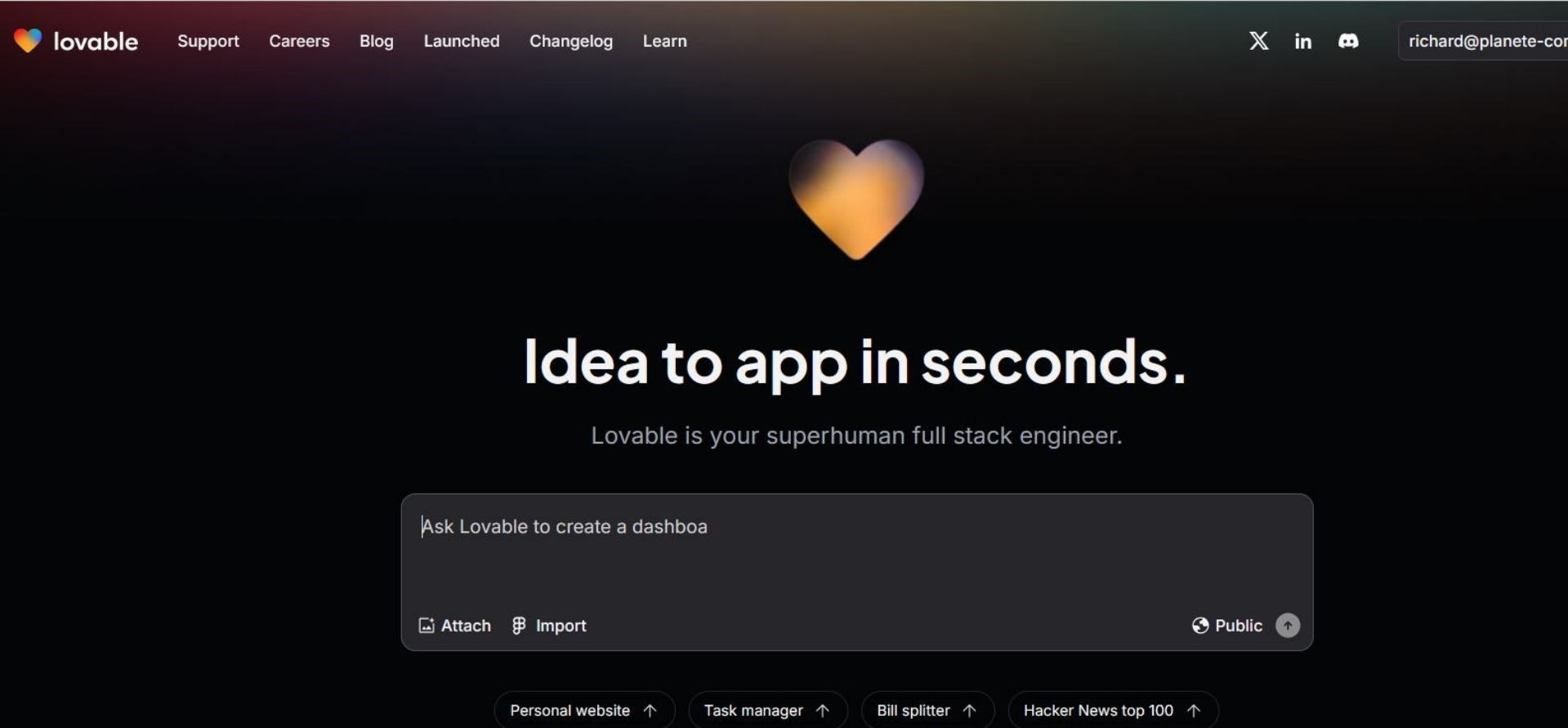
The code editor shows a preview of the 'Analytics Insight' page. The page title is 'Analytics Insight' and the subtitle is 'Upload your Google Analytics data for AI-powered analysis and insights'. Below the title, there is a section titled 'Upload Analytics Data' with the instruction 'Upload your Google Analytics monthly export file (JSON or CSV format)'. A dashed box indicates the area for file upload, with the text 'Drag and drop your file here or click to browse'. A note below the box states 'Maximum file size: 5MB'. A 'Connect to GitHub' button is visible in the top right corner of the editor.

Pour récupérer le code, connectez votre Lovable à votre compte Github


Cliquez sur Github

The image shows a composite screenshot of a development workflow. On the left, a chat interface displays a conversation about fixing HTML/CSS issues in a component named 'UploadForm.tsx'. The chat messages include a list of problems (missing closing tags, mismatched structure) and a solution (replacing CSS classes and fixing syntax). Below the chat is a 'Refactor UploadForm.tsx' button. In the center, a browser window shows the GitHub login page for the user 'lovable.dev', with a 'Sign in to GitHub' button and a 'Connect to GitHub' button. On the right, a toolbar contains icons for 'Supabase', 'GitHub', and 'Publish'. A blue arrow points from the 'Cliquez sur Github' text above to the GitHub icon in the toolbar.

Pour terminer, deux ou trois choses que l'on ne vous dit pas sur Lovable...






lovable Support Careers Blog Launched Changelog Learn X in richard@planete-con







Idea to app in seconds.

Lovable is your superhuman full stack engineer.

Ask Lovable to create a dashboa

 Attach  Import Public 

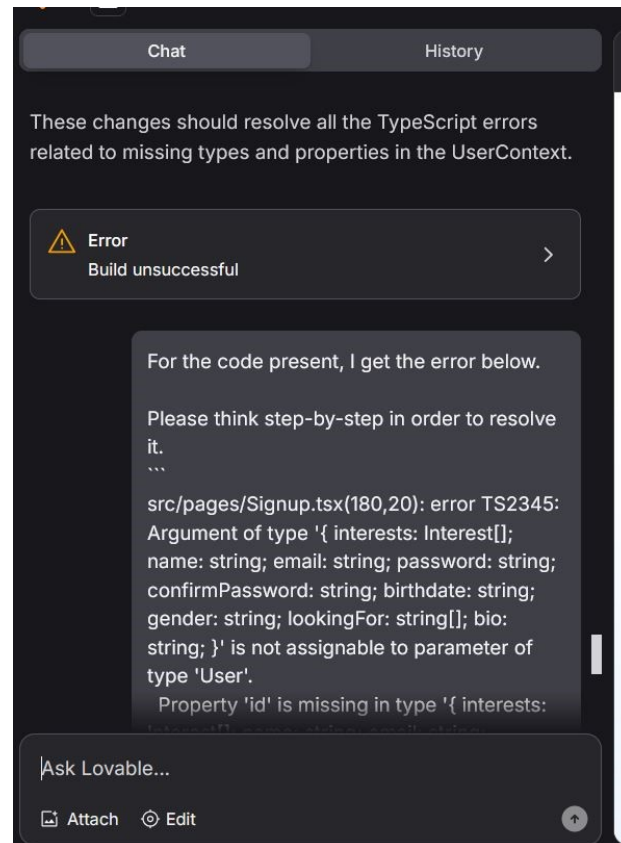
Personal website  Task manager  Bill splitter  Hacker News top 100 



Je vous ai un peu menti...

L'application précédente n'a pas fonctionné du premier coup car, en cours de génération, Lovable s'est bloqué, un peu comme lorsque ChatGPT « bugue »

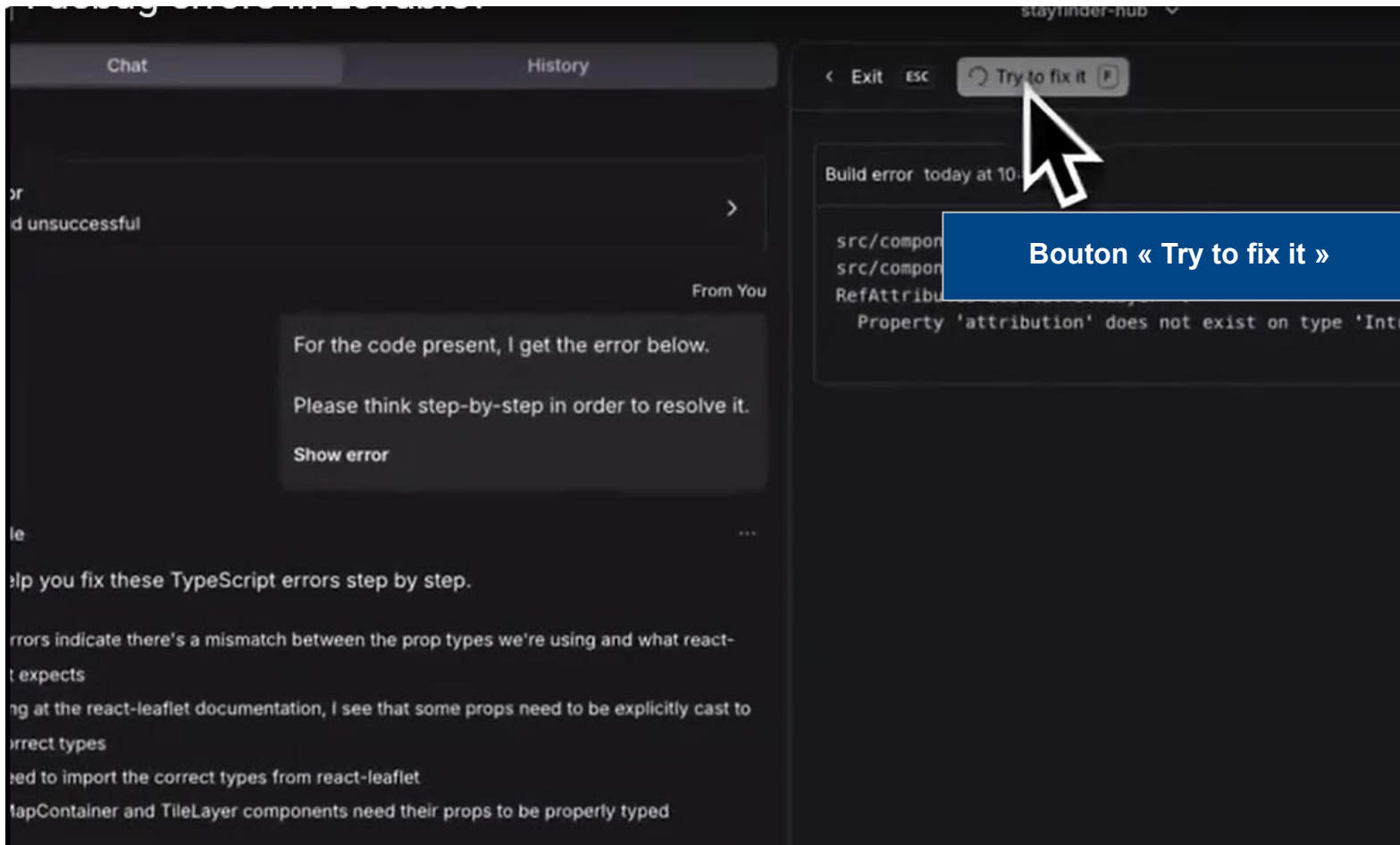
Tout à coup, Lovable peut cesser, même pour une application simple, de générer le code et vous envoyer des messages abscons...





Mais, heureusement, il y a un bouton magique

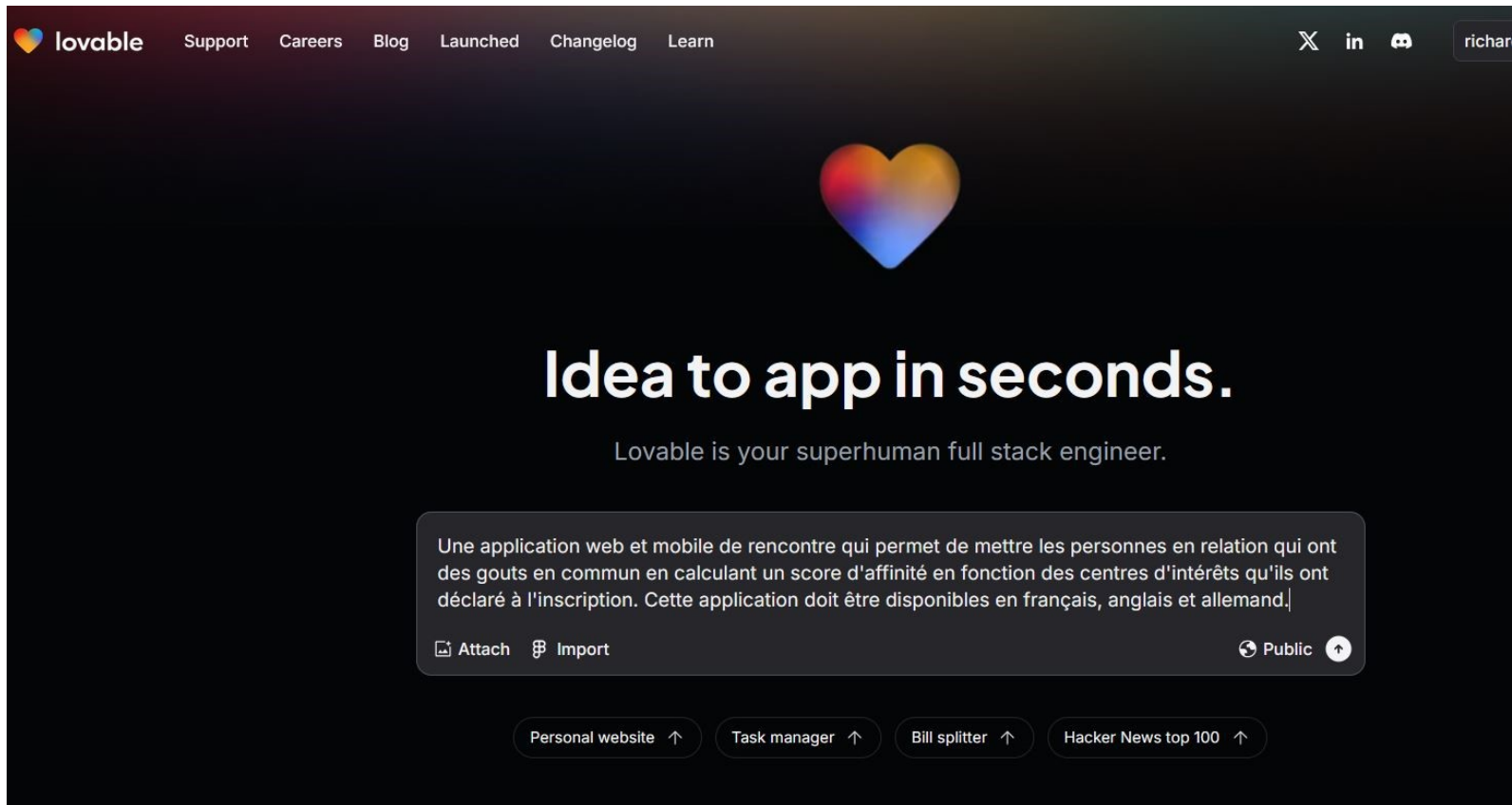
Il faut alors cliquer sur la box avec le message d'erreur dans la colonne de dialogue, puis cliquer sur le bouton « Try to fix it » dans la zone de travail à droite et croiser les doigts...





Et, parfois, Lovable accouche de l'application
dans la douleur

J'ai, ainsi, essayé de générer une application plus complexe de rencontre par affinité avec calcul de score de comptabilité, multilingue qui plus est



The screenshot shows the Lovable website interface. At the top, there is a navigation bar with the Lovable logo and links for Support, Careers, Blog, Launched, Changelog, and Learn. On the right, there are social media icons for X, LinkedIn, and a user profile for 'richar'. The main content area features a large, colorful heart icon. Below the heart, the text reads 'Idea to app in seconds.' followed by 'Lovable is your superhuman full stack engineer.' A text box contains the following description: 'Une application web et mobile de rencontre qui permet de mettre les personnes en relation qui ont des goûts en commun en calculant un score d'affinité en fonction des centres d'intérêts qu'ils ont déclaré à l'inscription. Cette application doit être disponibles en français, anglais et allemand.' Below the text box are buttons for 'Attach', 'Import', and 'Public'. At the bottom, there are four buttons for other projects: 'Personal website', 'Task manager', 'Bill splitter', and 'Hacker News top 100'.

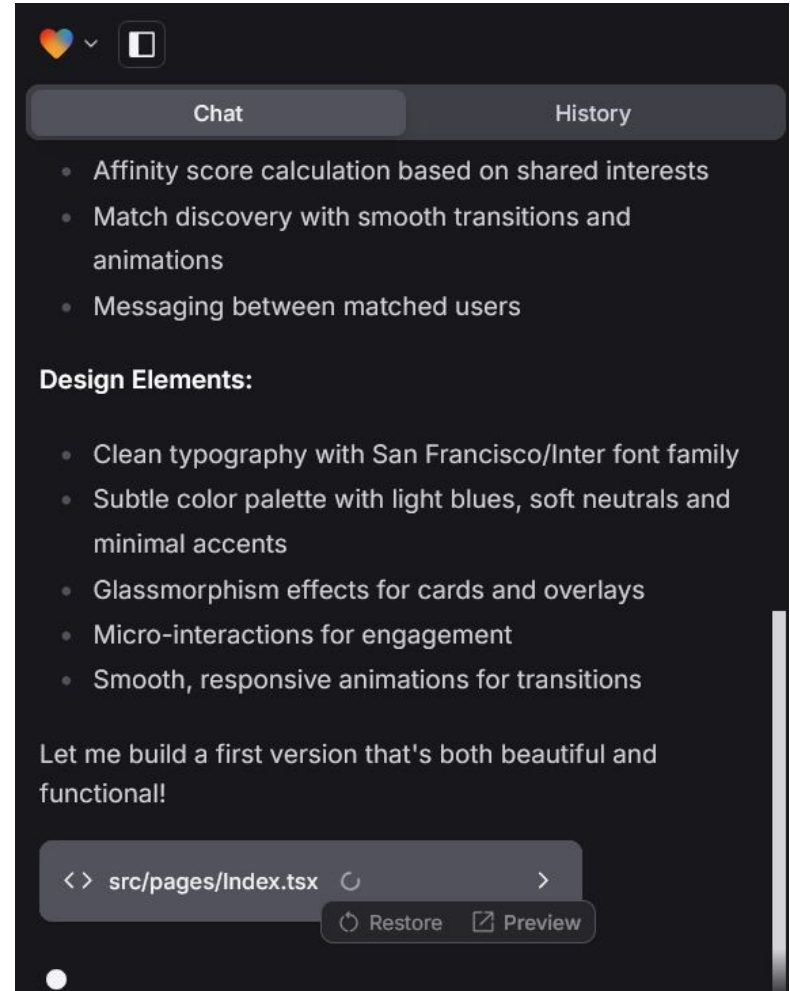
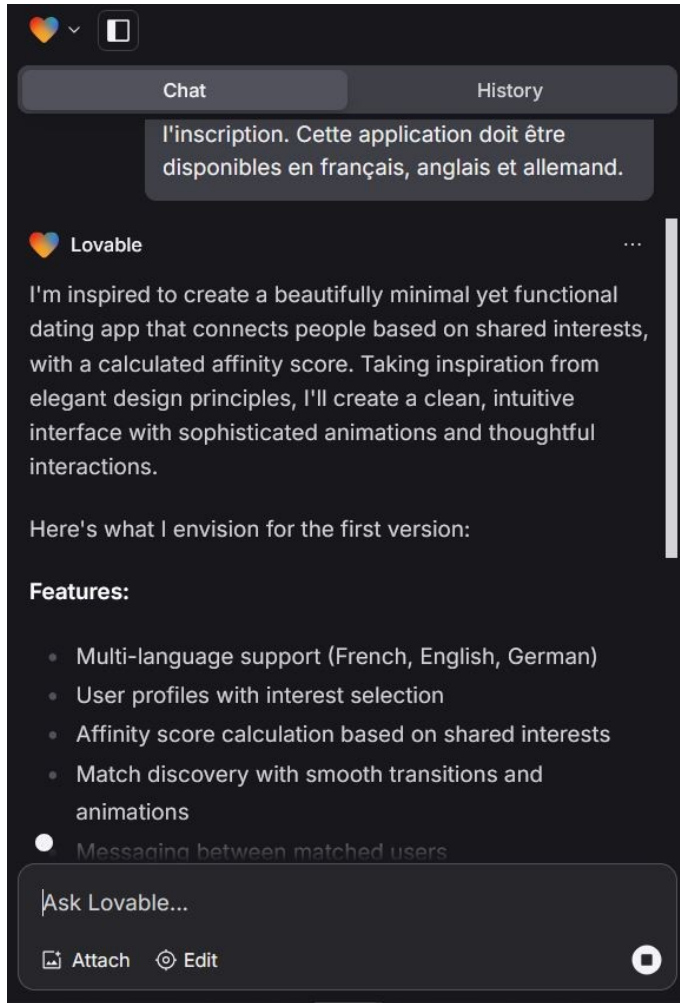
Au début, tout allait bien

The screenshot shows a web application interface for 'affinity-finder'. The browser address bar displays 'preview--affinity-finder.lovable.app'. The page features a large grey heart icon and the text 'Spinning up preview...'. Below this, there are four instructions:

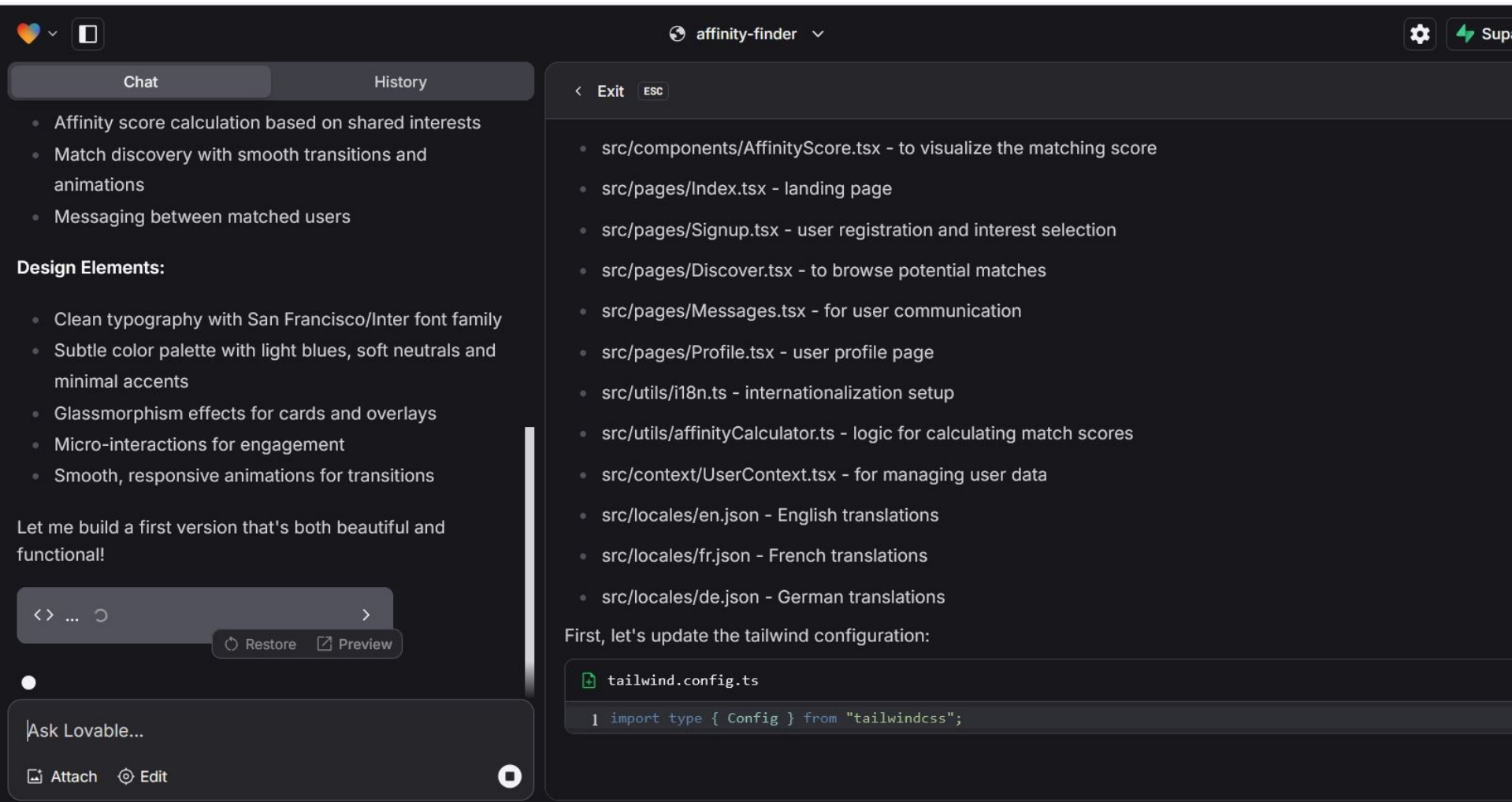
- Set custom knowledge for every edit
- Connect Supabase for backend
- Collaborate at source, via GitHub
- Deploy when you're ready

On the left side, there is a chat window with a 'History' tab. A message in the chat reads: 'Une application web et mobile de rencontre qui permet de mettre les personnes en relation qui ont des goûts en commun en calculant un score d'affinité en fonction des centres d'intérêts qu'ils ont déclaré à l'inscription. Cette application doit être disponibles en français, anglais et allemand.'

Lovable interprétait mon intention correctement



Lovable enchaînait avec l'inventaire des fichiers à créer



The screenshot displays the Lovable AI IDE interface. On the left, a chat window is open with a 'Chat' tab selected. The chat history shows a list of bullet points describing the project's features and design elements. Below the chat history, there is a text input field with the prompt 'Let me build a first version that's both beautiful and functional!' and a 'Restore' button. At the bottom of the chat window, there is a text input field with the prompt 'Ask Lovable...' and 'Attach' and 'Edit' buttons.

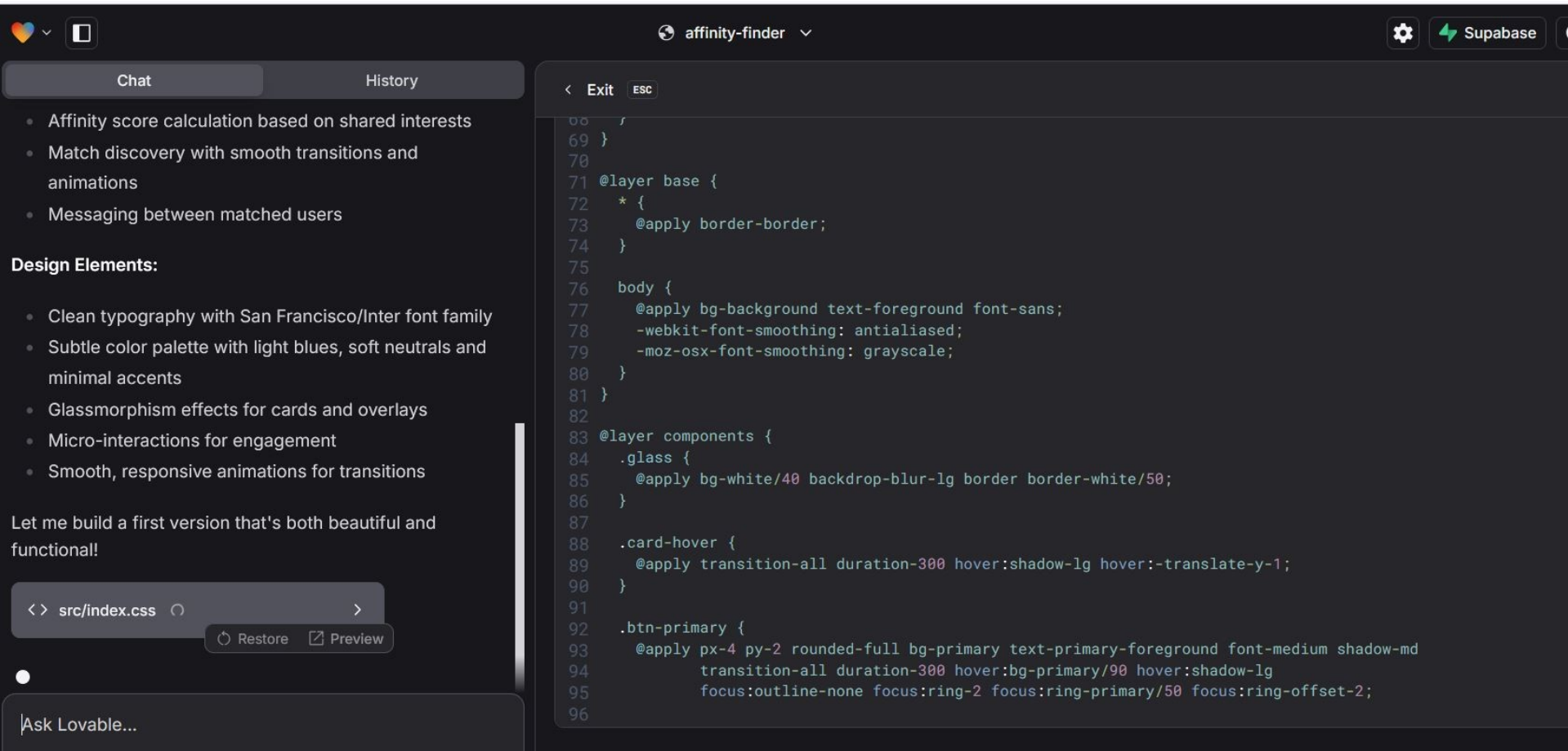
The main code editor on the right shows a file tree for 'affinity-finder'. The file list includes:

- src/components/AffinityScore.tsx - to visualize the matching score
- src/pages/Index.tsx - landing page
- src/pages/Signup.tsx - user registration and interest selection
- src/pages/Discover.tsx - to browse potential matches
- src/pages/Messages.tsx - for user communication
- src/pages/Profile.tsx - user profile page
- src/utils/i18n.ts - internationalization setup
- src/utils/affinityCalculator.ts - logic for calculating match scores
- src/context/UserContext.tsx - for managing user data
- src/locales/en.json - English translations
- src/locales/fr.json - French translations
- src/locales/de.json - German translations

Below the file list, the text 'First, let's update the tailwind configuration:' is displayed. The code editor shows the following code snippet for 'tailwind.config.ts':

```
1 import type { Config } from "tailwindcss";
```

Lovable génère le code des différents fichiers, les uns après les autres...



The screenshot shows a development environment with a chat interface on the left and a code editor on the right. The chat interface displays a list of features and design elements, and a prompt to build a first version. The code editor shows CSS code for a project named 'affinity-finder'.

Chat

- Affinity score calculation based on shared interests
- Match discovery with smooth transitions and animations
- Messaging between matched users

Design Elements:

- Clean typography with San Francisco/Inter font family
- Subtle color palette with light blues, soft neutrals and minimal accents
- Glassmorphism effects for cards and overlays
- Micro-interactions for engagement
- Smooth, responsive animations for transitions

Let me build a first version that's both beautiful and functional!

src/index.css

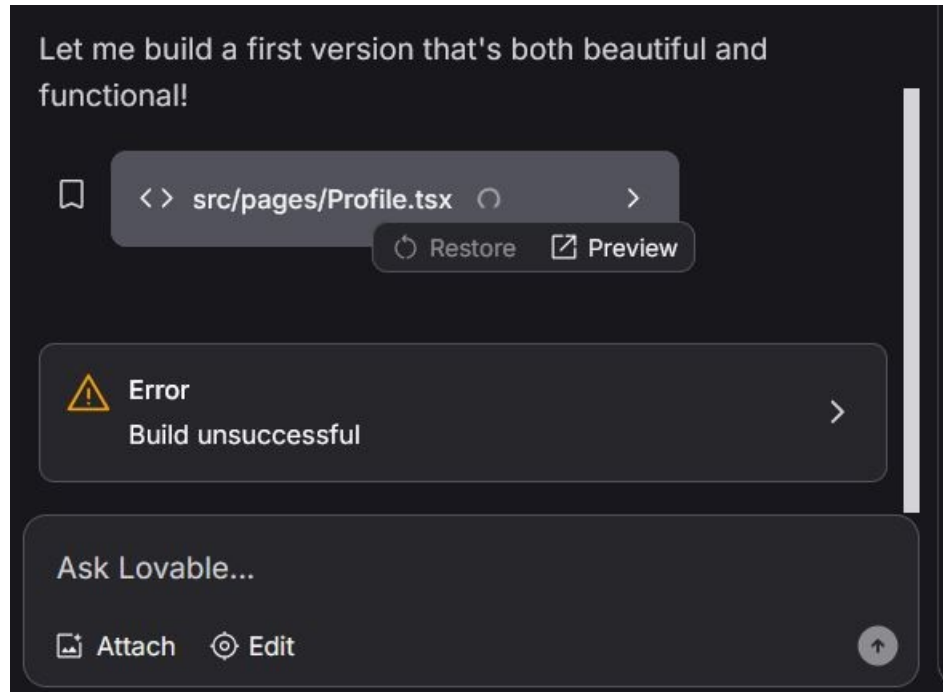
Restore Preview

Ask Lovable...

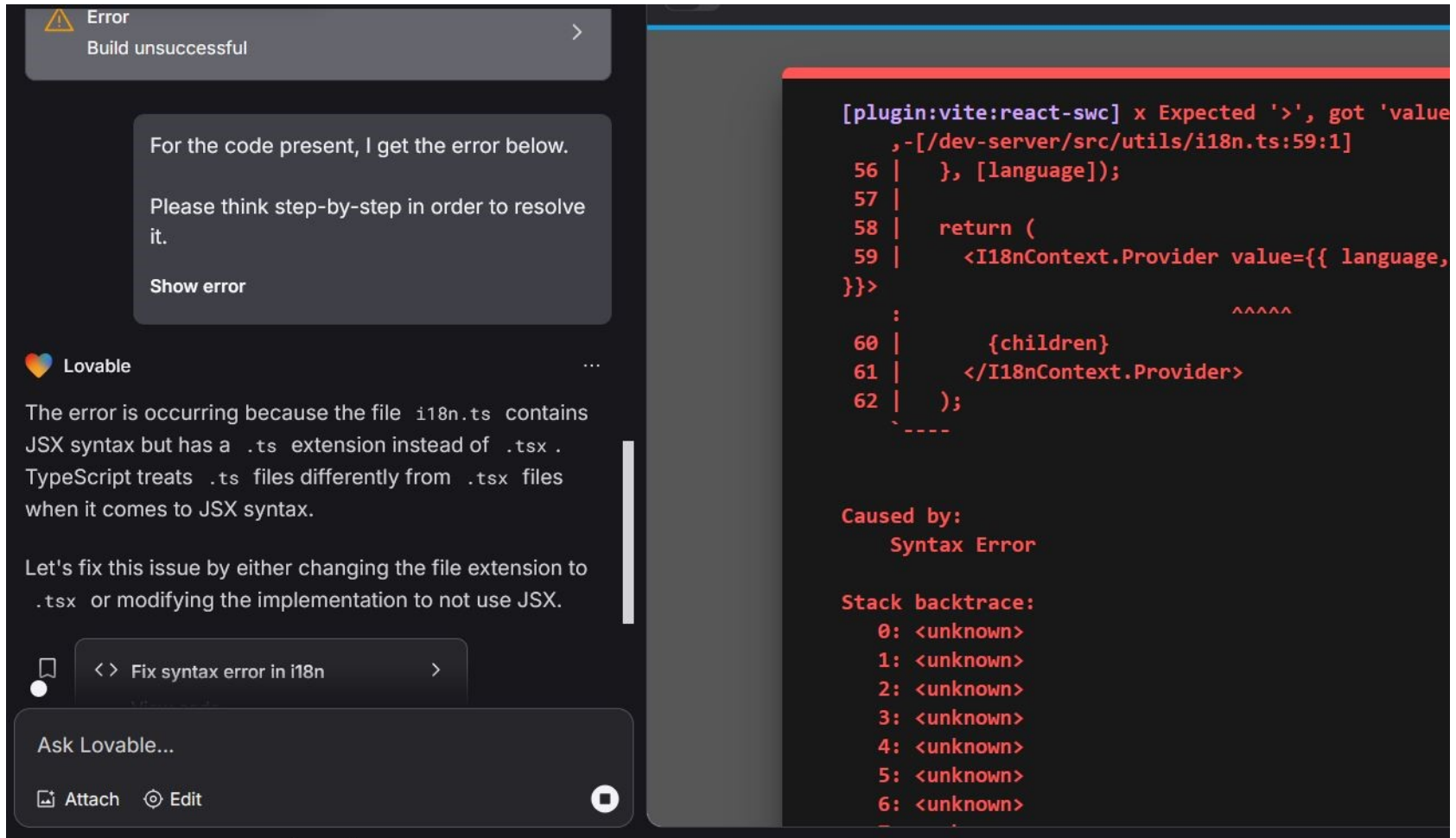
Code Editor

```
< Exit ESC
68 }
69 }
70
71 @layer base {
72   * {
73     @apply border-border;
74   }
75
76   body {
77     @apply bg-background text-foreground font-sans;
78     -webkit-font-smoothing: antialiased;
79     -moz-osx-font-smoothing: grayscale;
80   }
81 }
82
83 @layer components {
84   .glass {
85     @apply bg-white/40 backdrop-blur-lg border border-white/50;
86   }
87
88   .card-hover {
89     @apply transition-all duration-300 hover:shadow-lg hover:-translate-y-1;
90   }
91
92   .btn-primary {
93     @apply px-4 py-2 rounded-full bg-primary text-primary-foreground font-medium shadow-md
94     transition-all duration-300 hover:bg-primary/90 hover:shadow-lg
95     focus:outline-none focus:ring-2 focus:ring-primary/50 focus:ring-offset-2;
96 }
```

Et puis au bout de 5 minutes, gros bug sans trop d'explication



En cliquant sur la boîte d'erreur, je vois du code «rouge»
...sans comprendre ce que je pouvais faire



The screenshot shows an IDE interface with an error notification and a code editor. The error notification on the left contains the following text:

Error
Build unsuccessful

For the code present, I get the error below.

Please think step-by-step in order to resolve it.

Show error

Lovable

The error is occurring because the file `i18n.ts` contains JSX syntax but has a `.ts` extension instead of `.tsx`. TypeScript treats `.ts` files differently from `.tsx` files when it comes to JSX syntax.

Let's fix this issue by either changing the file extension to `.tsx` or modifying the implementation to not use JSX.

<> Fix syntax error in i18n

Ask Lovable...

Attach Edit

The code editor on the right shows the following code snippet:

```
[plugin:vite:react-swc] x Expected '>', got 'value',
-[/dev-server/src/utils/i18n.ts:59:1]
56 |   }, [language]);
57 |
58 |   return (
59 |     <I18nContext.Provider value={{ language,
}}>
:
60 |       {children}
61 |     </I18nContext.Provider>
62 |   );
  -----
```

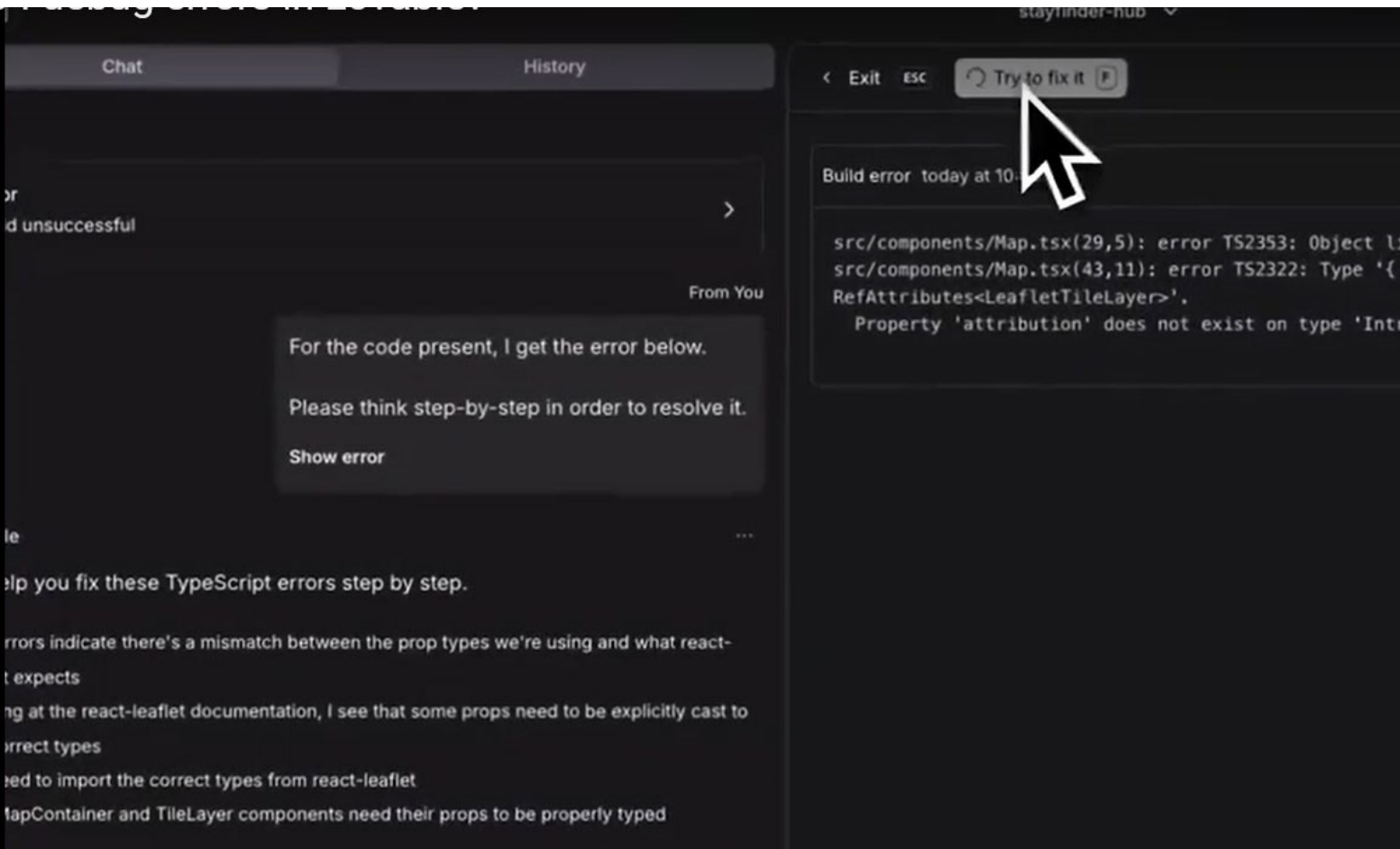
Caused by:
Syntax Error

Stack backtrace:

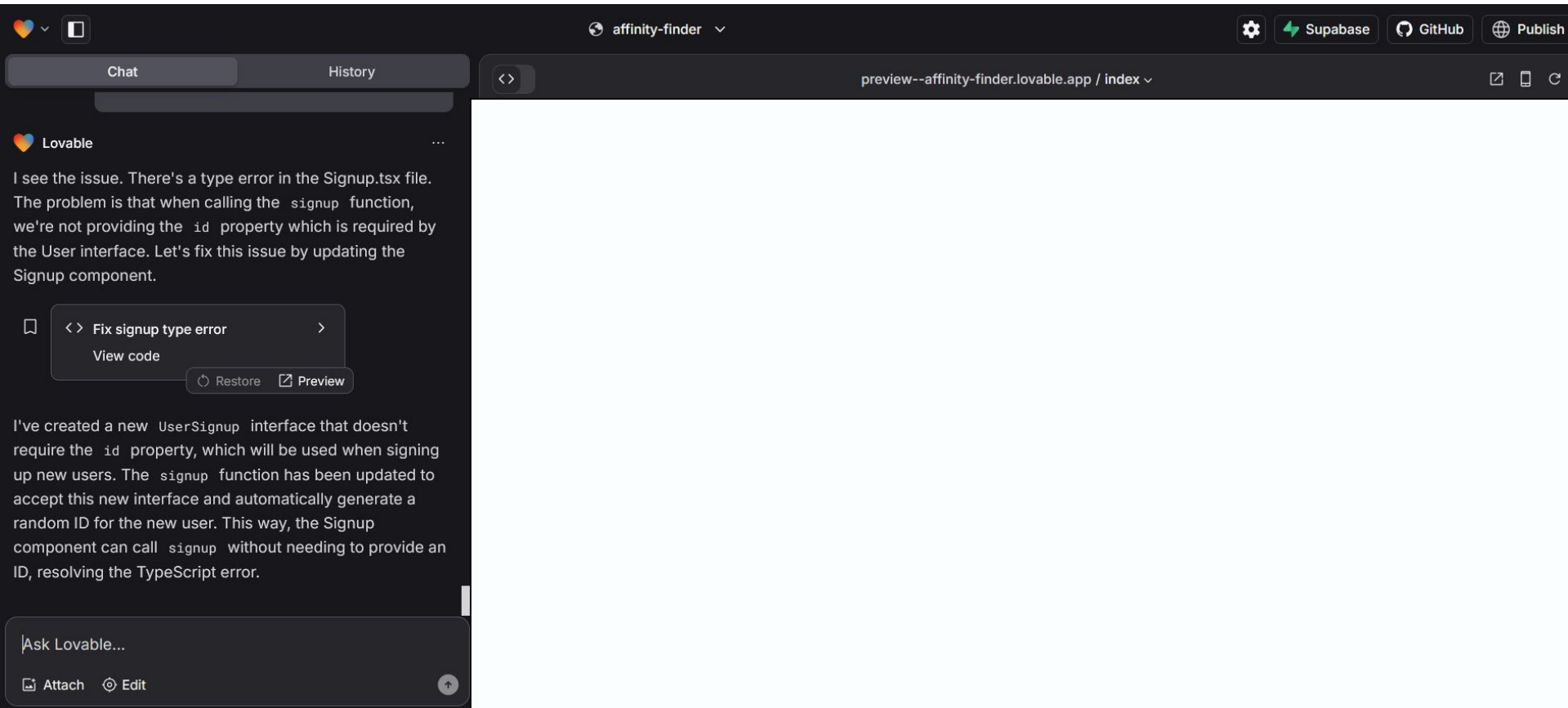
- 0: <unknown>
- 1: <unknown>
- 2: <unknown>
- 3: <unknown>
- 4: <unknown>
- 5: <unknown>
- 6: <unknown>

55

Mais, en m'acharnant sur le bouton « Try to fix it », Lovable a finir par se remettre en marche, mais j'ai bien cru qu'il n'y arriverait pas...



D'ailleurs, lorsqu'il a eu terminé de générer le code, j'ai eu le droit à une page blanche en guise de home page



The screenshot displays a development environment with a chat interface on the left and a browser preview on the right. The chat window, titled 'Chat', shows a message from 'Lovable' explaining a TypeScript error in the `Signup.tsx` file. The message states: "I see the issue. There's a type error in the Signup.tsx file. The problem is that when calling the `signup` function, we're not providing the `id` property which is required by the User interface. Let's fix this issue by updating the Signup component." Below the message is a code diff snippet titled "Fix signup type error" with a "View code" link and "Restore" and "Preview" buttons. The message continues: "I've created a new `UserSignup` interface that doesn't require the `id` property, which will be used when signing up new users. The `signup` function has been updated to accept this new interface and automatically generate a random ID for the new user. This way, the Signup component can call `signup` without needing to provide an ID, resolving the TypeScript error." The browser preview on the right shows a blank white page at the URL `preview--affinity-finder.lovable.app / index`. The top navigation bar includes links for 'Supabase', 'GitHub', and 'Publish'.

J'ai, donc, abandonné

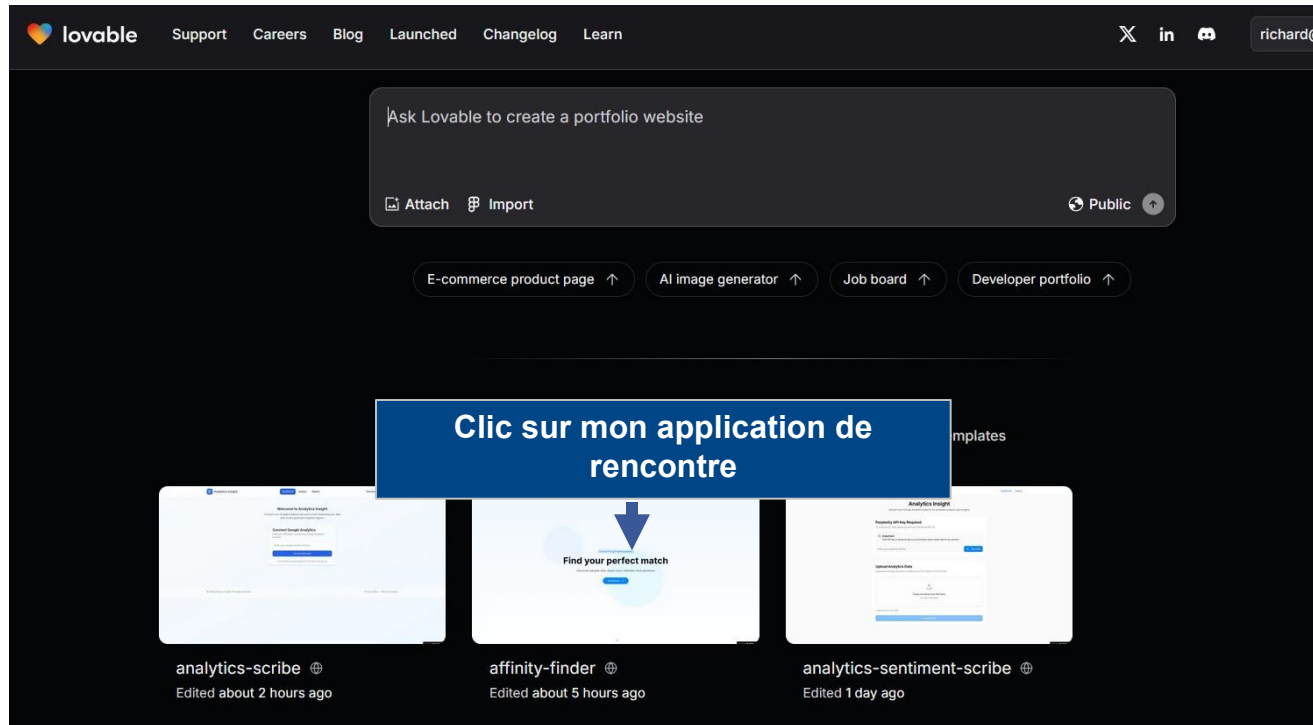


Et, deux heures plus tard...

Miracle !



J'ai relancé l'application à partir du « tableau de bord »



Et hop, ma home page était là !

The screenshot shows a development environment with a chat window on the left and a web preview on the right. The chat window is titled "Lovable" and contains a message from the AI assistant:

I see the issue. There's a type error in the Signup.tsx file. The problem is that when calling the `signup` function, we're not providing the `id` property which is required by the User interface. Let's fix this issue by updating the Signup component.

Below the message is a code editor snippet:

```
<> Fix signup type error >
View code
Restore Preview
```

The chat window also contains a follow-up message:

I've created a new `UserSignup` interface that doesn't require the `id` property, which will be used when signing up new users. The `signup` function has been updated to accept this new interface and automatically generate a random ID for the new user. This way, the Signup component can call `signup` without needing to provide an ID, resolving the TypeScript error.

The web preview on the right shows the home page of the "affinity-finder" application. The page has a light blue background with a large white circle in the center. The main heading is "Trouvez votre match parfait" (Find your perfect match). Above the heading is a button that says "Connectez-vous par les passions partagées" (Connect via shared passions). Below the heading is the text "Découvrez des personnes qui partagent vos intérêts et passions" (Discover people who share your interests and passions). At the bottom of the circle is a blue button that says "Commencer" (Start) with a right arrow. The browser address bar shows "preview--affinity-finder.lovable.app / index".

Lovable, une application en devenir

Vous l'avez compris, Lovable est une jeune application, plein de promesse, européenne de surcroît, mais pas parfaite...

En même temps, je me suis laissé dire que le leader mondial de l'édition logicielle, qui fête son 50^{ème} anniversaire en 2025, sortait encore des logiciels bien imparfaits...

Lovable, une application à dompter... et à prompter

Comme je l'ai déjà évoqué, si vous souhaitez obtenir l'application que vous avez en tête, il faut préciser ce que vous avez en tête à Lovable:

- ce que l'on souhaite que l'application fasse:
 - en front office (ce que voit l'utilisateur)
 - en back office (ce que peut faire l'administrateur)
- Vos préférences en matière de design graphique
- Les process sous jacents (par exemple, les emails transactionnels qui doivent être envoyés)
- Les algorithmes que vous imaginez (comme le système de calcul du score de comptabilité du site de rencontre dont j'ai parlé)
- Vos préférences techniques si vous en avez (par exemple, je demande toujours à mes développeurs de sortir du code tous les textes qui doivent s'afficher à l'écran pour les basculer dans des dictionnaires, afin de faciliter le lancement de versions multilingues)
- La plateforme technique sur laquelle vous hébergerez l'application lorsque Lovable l'aura créée...

Lovable, une application à dompter... ... avec un hack ?

En réalité, vous devez rédiger un cahier des charges, ce qui nécessite que vous vous asseyiez un moment pour réfléchir au fonctionnement détaillé de votre application.

Mais que doit contenir un cahier des charges ?

Mon hack, si vous ne savez pas trop rédiger un cahier des charges...

... demandez à ChatGPT !

Le mauvais hack

J'ai vu sur internet des vidéos où l'on expliquait qu'il suffisait de demander à ChatGPT de rédiger le cahier des charges « détaillé » « pour Lovable » d'une application qui fait ceci..

Que renvoie ChatGPT dans ce cas ?



Un cahier des charges très superficiel qui tient en une page !

Un vrai cahier des charges, c'est plus détaillé que cela.

Mon hack pour le cahier des charges

Pour obtenir un cahier plus substantiel, voici un prompt très précis
« Je désire *créer une application de rencontre par affinités qui mettent en relation des personnes en fonction des points communs qu'ils ont au niveau des loisirs, des préférences culturelles ou des projets de vie* [à remplacer par la description de votre application]. Je dois rédiger un cahier des charges technique pour un développeur, mais je ne sais pas comment rédiger ce cahier des charges. Pourrais tu rédiger un cahier des charges très détaillé, totalement rédigé, mais en m'indiquant pour chaque point ce sur quoi je dois exprimer mes préférences en proposant à chaque fois des exemples de choix possibles. Tu insisteras sur les points que tu estimeras sensibles parce qu'ils auront des conséquences techniques importantes. »

Pour obtenir un cahier des charges de 4 ou 5 pages, **vous devez impérativement l'adresser à un modèle dit « de raisonnement »**, comme ChatGPT o1, Perplexity Raisonnement ou Deepseek / Deepresearch »

Les tarifs

Comme de nombreux services d'IA, Lovable propose des abonnements donnant droit à un certain nombre de crédit, qui sont un peu difficile à comprendre.

Voici ce que j'en ai compris de cette tarification.

Version gratuite: 3 applications maximum, 5 prompts par jour

Starter (\$20/mois) : 100 prompts par mois

Launch (\$50/mois) : 250 prompts par mois

Scale (\$100 à \$900/mois): 500 à 5000 prompts par mois

Merci !

Raphaël Richard

Les pépites de l'Intelligence artificielle
sur [Linkedin.com/in/raphaelric](https://www.linkedin.com/in/raphaelric)

Raphael.richard@neodia.com

06 62 55 00 10



Besoin de former vos
équipes sur l'IA
et l'IA générative et de
développer des applis d'IA

Nous sommes là

Neodia
24pm Academy

Formations IA
Conseil IA
AMAO IA

👤 06 62 55 00 10

📱 Raphaël Richard

✉️ Raphael.richard@neodia.com

🌐 <https://24pm.Com>