Here is the **complete Lovable prompt series**—combined into a single, step-by-step set of instructions you can paste into Lovable to generate your **Twilio-connected meme messaging app**, including database integration, ML-based categorization, and dashboard functionality.

**💬 Unified Lovable Prompt: Meme Messaging App with Twilio and AI Categorization**

**🧠 Goal**

Build a secure app that sends personalized memes via SMS to users based on their interests, using Twilio and a meme database. Retrieve the user’s phone number and preferences from a user database, select a meme using keyword or ML matching, and send it automatically.

**🧩 Prompts to Paste in Lovable (Sequentially or as a Bundle)**

**🔐 1. Twilio Integration**

text

CopyEdit

Create a secure integration with Twilio’s SMS API using environment variables:

- TWILIO\_ACCOUNT\_SID

- TWILIO\_AUTH\_TOKEN

- TWILIO\_PHONE\_NUMBER

Authenticate and allow sending outbound text messages using Twilio from within the app.

**🗃️ 2. User Database Retrieval**

text

CopyEdit

Connect to a SQL database with a table named `users`.

Select all users where `subscribed\_to\_texts = true` and retrieve their `phone\_number`, `name`, and `interests` fields.

Table structure:

text

CopyEdit

users (id, name, phone\_number, subscribed\_to\_texts, interests)

**🖼️ 3. Meme Selection by Category**

text

CopyEdit

Connect to a `memes` table that includes:

- id

- category

- image\_url

- description

Create a function that selects one meme that matches a user’s interest based on keyword or category.

If multiple memes match, select one at random.

**🧠 4. (Optional) ML Categorization Engine**

text

CopyEdit

Create a basic machine learning classifier that uses a user’s stated interests to predict the most relevant meme category.

Use this to improve matching between users and meme categories over time.

You can use a simple keyword-weighted vector model or scikit-learn-based classifier.

**📤 5. Send Meme via SMS**

text

CopyEdit

Create a function that sends the selected meme’s description and image URL to each user’s phone number via Twilio SMS.

Sample message format:

"Here’s a meme to brighten your day 😎 – {description} – {image\_url}"

**🔁 6. Automate Delivery**

text

CopyEdit

Schedule this flow to run daily at 9 AM.

It should:

- Retrieve all subscribed users

- Match a meme to their interests

- Send it by SMS

- Log the status in an `sms\_logs` table

sms\_logs schema:

text

CopyEdit

sms\_logs (id, user\_id, message\_status, timestamp)

**📊 7. Admin Dashboard UI**

text

CopyEdit

Create an admin dashboard with the following:

- Table showing user info and subscription status

- Table of all memes with filters by category

- Button to trigger manual SMS send

- Bar chart showing number of messages sent by day

**🛑 8. Opt-Out UI**

text

CopyEdit

Create a public page that allows a user to unsubscribe from SMS by entering their phone number.

When submitted, it sets `subscribed\_to\_texts = false` in the `users` table.

**✅ Final Notes for Lovable:**

* Use Python for the backend.
* Use React (or Flutter) for the mobile/web frontend.
* Use secure encryption (AES-256) and HTTPS for all user data.
* Ensure Twilio integration handles delivery failures and logs them.

Would you like me to bundle this into a downloadable .txt file or generate the full backend API logic in Python to accompany it?

**You said:**

both

**ChatGPT said:**

✅ All set! You can now download the following files:

1. 📄 Lovable\_MemeApp\_Twilio\_Prompts.txt – A full Lovable prompt script to build your Twilio + meme app.
2. 🐍 meme\_sms\_backend.py – A working Python backend script that connects to Twilio, queries users, fetches memes, and sends SMS.

Let me know if you’d like a version that works with PostgreSQL, includes webhook handling, or adds image attachments via MMS. ​​