LiveKit Set Up - Basic

by Henryk Brzozowski

Free Voice AI course: https://www.skool.com/voice-ai-bootcamp

Pre-requisites

- Have basic Cursor or IDE experience
- An IDE (Visual Studio Code or Cursor)
- v3.11 of Python (python.org/downloads) | For Mac users you can also download from there or run `brew install python@3.11`. If you have 3.10, 3.11 or 3.12 will work fine.
- Deepgram, OpenAI and Cartesia accounts:
  - Deepgram **Enterprise Voice AI: STT, TTS & Agent APIs | Deepgram** (STT)
  - **OpenAI Platform** (Our LLM)
  - **Cartesia - The fastest, ultra-realistic voice AI platform.** (TTS)

Skip to content ↪

▾ Initial Set Up

1. In your IDE (In my case Cursor), run each one of these commands one by one to start up a virtual environment, this will is where all the “pip” installs will go (I think of them as python install packages):

```
# Initiate python -m venv venv # Activate venv\Scripts\activate # Mac activate source venv/bin/activate # Turn off your venv deactivate
```

Note: ensure the root folder of your project is being used before running this

3. Then install the initial packages

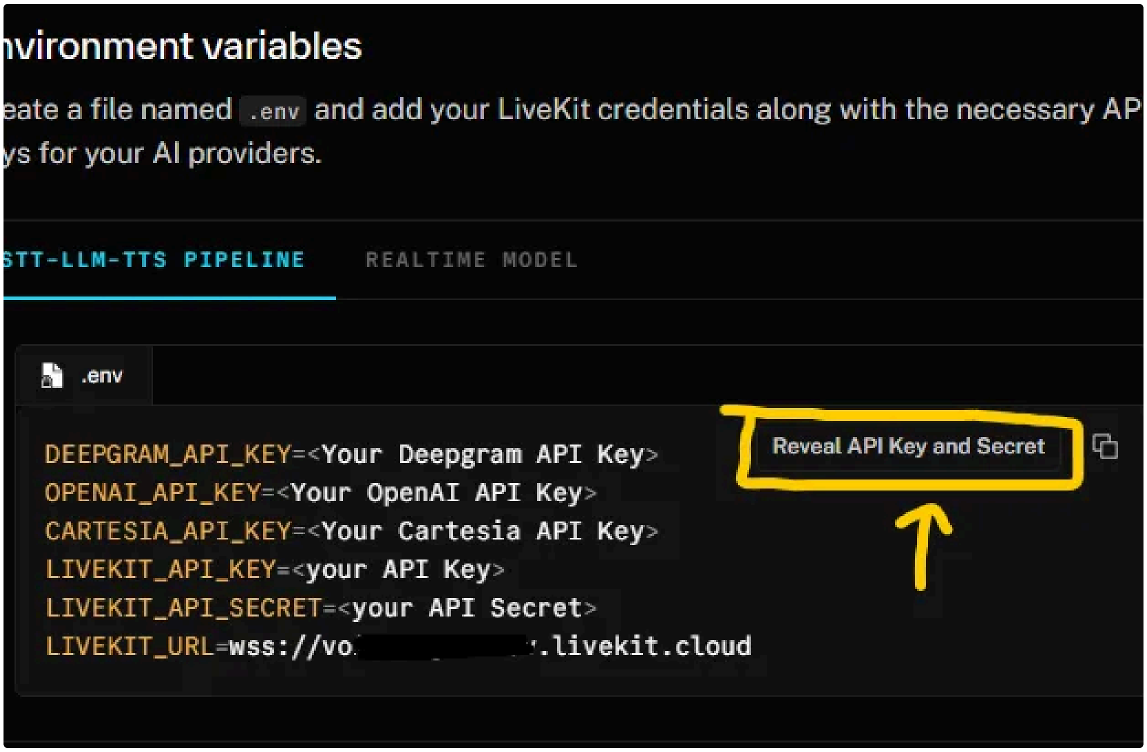
```
pip install "livekit-agents[deepgram,openai,cartesia,silero,turn-detector]~=1.0" "livekit-plugins-noise-cancellation~=0.2" "python-dotenv"
```

Skip to content ↪

4. create a .env with:

```
DEEPGRAM_API_KEY=<Your Deepgram API Key> OPENAI_API_KEY=<Your OpenAI API Key> CARTESIA_API_KEY=<Your Cartesia API Key> LIVEKIT_API_KEY=<your API Key> LIVEKIT_API_SECRET=<your API Secret> LIVEKIT_URL=<LK URL>
```

Note: you can find some of them by clicking on the Reveal API Key and Secret Button once you hover over the .env area in the LiveKit docs: https://docs.livekit.io/agents/start/voice-ai/



nvironment variables

eate a file named `.env` and add your LiveKit credentials along with the necessary AP ys for your AI providers.

```
DEEPGRAM_API_KEY=<Your Deepgram API Key>
OPENAI_API_KEY=<Your OpenAI API Key>
CARTESIA_API_KEY=<Your Cartesia API Key>
LIVEKIT_API_KEY=<your API Key>
LIVEKIT_API_SECRET=<your API Secret>
LIVEKIT_URL=wss://vo[redacted].livekit.cloud
```

Skip to content

5. Copy this code into the agent.py file. Here we have two options:

▼ a) Multilingual

```
from dotenv import load_dotenv from livekit import agents from liv
ekit.agents import AgentSession, Agent, RoomInputOptions from live
kit.plugins import ( openai, cartesia, deepgram, noise_cancellatio
n, silero, ) from livekit.plugins.turn_detector.multilingual impor
t MultilingualModel load_dotenv() class Assistant(Agent): def __in
it__(self) -> None: super().__init__(instructions="You are a helpf
ul voice AI assistant.") async def entrypoint(ctx: agents.JobConte
xt): session = AgentSession( stt=deepgram.STT(model="nova-3", lang
uage="multi"), llm=openai.LLM(model="gpt-4o-mini"), tts=cartesia.T
TS(model="sonic-2", voice="f786b574-daa5-4673-aa0c-cbe3e8534c02"),
vad=silero.VAD.load(), turn_detection=MultilingualModel(), ) await
session.start( room=ctx.room, agent=Assistant(), room_input_option
s=RoomInputOptions( # LiveKit Cloud enhanced noise cancellation #
- If self-hosting, omit this parameter # - For telephony applicati
ons, use `BVCTelephony` for best results noise_cancellation=noise_
cancellation.BVC(), ), ) await ctx.connect() await session.generat
e_reply( instructions="Greet the user and offer your assistance."
) if __name__ == "__main__": agents.cli.run_app(agents.WorkerOptio
ns(entrypoint_fnc=entrypoint))
```

Skip to content ↪

b) English

```
from dotenv import load_dotenv from livekit import agents from liv
ekit.agents import AgentSession, Agent, RoomInputOptions from live
kit.plugins import ( openai, cartesia, deepgram, noise_cancellatio
n, silero, ) from livekit.plugins.turn_detector.english import Eng
lishModel # from livekit.plugins.turn_detector.multilingual import
MultilingualModel for multiple languages load_dotenv() class Assis
tant(Agent): def __init__(self) -> None: super().__init__(instruct
ions="You are a helpful voice AI assistant.") async def entrypoint
(ctx: agents.JobContext): session = AgentSession( stt=deepgram.STT
(model="nova-3", language="en"), # language = "multi" for multiple

e ) await session.start( room=ctx.room, agent=Assistant(), room_in
put_options=RoomInputOptions( # LiveKit Cloud enhanced noise cance
llation # - If self-hosting, omit this parameter # - For telephony
applications, use `BVCTelephony` for best results noise_cancellati
on=noise_cancellation.BVC(), ), ) await ctx.connect() await sessio
n.generate_reply( instructions="Greet the user and offer your assi
stance." ) if __name__ == "__main__": agents.cli.run_app(agents.Wo
rkerOptions(entrypoint_fnc=entrypoint))
```

6. Try running this command:

```
python agent.py download-files
```

If issues come up when using the above command, a few things I used to trouble shoot on my windows:

1. pip install scipy
2. Install Visual C++ Redistributables and then restart computer

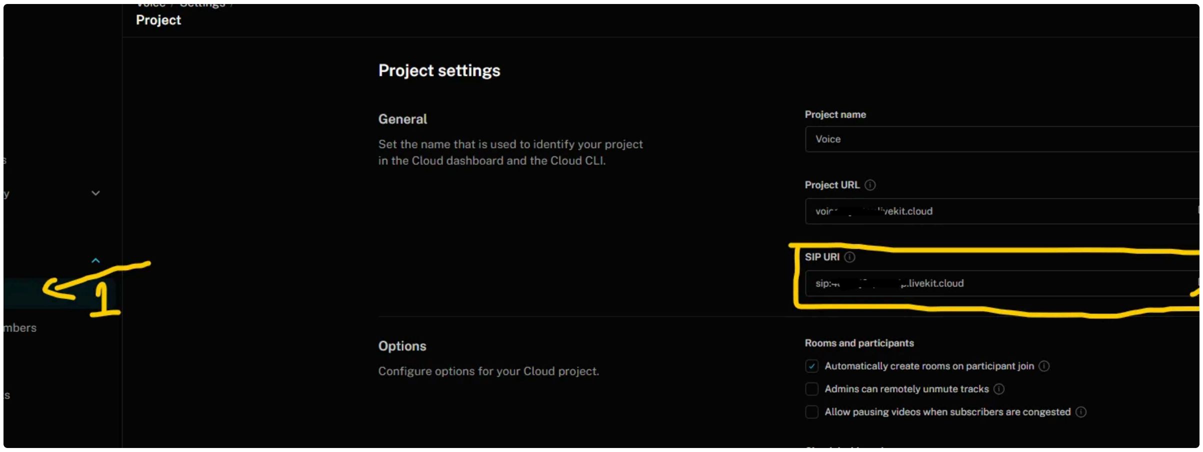
7. To start talking to your LiveKit agent

```
python agent.py console
```

Skip to content

Twilio Set Up (Inbound)

1. https://cloud.livekit.io/projects
2. Go into Settings → Project → Copy SIP URI



Project

Project settings

General

Set the name that is used to identify your project in the Cloud dashboard and the Cloud CLI.

Project name: Voice

Project URL: voi[redacted]vekit.cloud

SIP URI: sip:[redacted].livekit.cloud

Options

Configure options for your Cloud project.

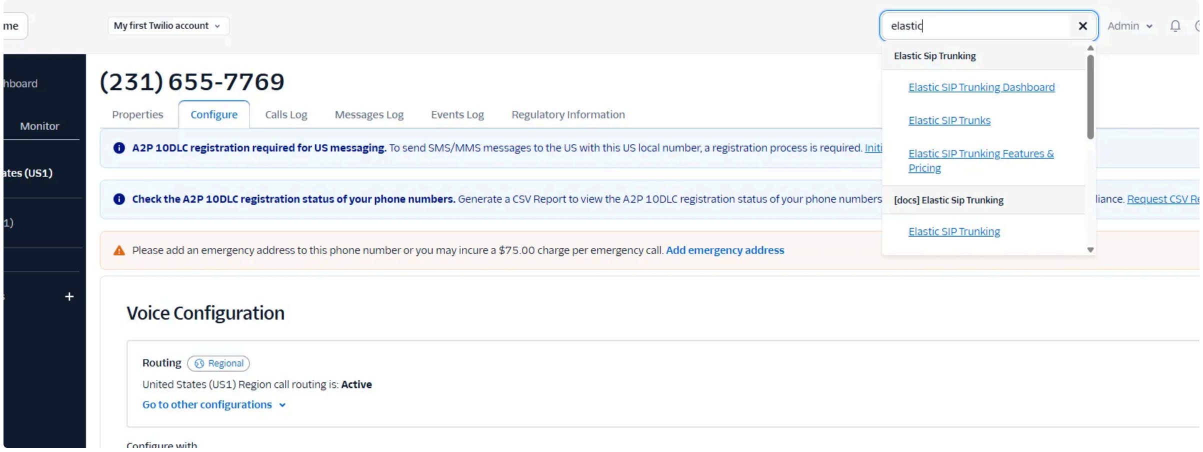
Rooms and participants

☑ Automatically create rooms on participant join

☐ Admins can remotely unmute tracks

☐ Allow pausing videos when subscribers are congested

3. Go to Twilio → Buy Number → Search Elastic → Elastic SIP Trunks



My first Twilio account

elastic

(231) 655-7769

A2P 10DLC registration required for US messaging. To send SMS/MMS messages to the US with this US local number, a registration process is required.

Check the A2P 10DLC registration status of your phone numbers. Generate a CSV Report to view the A2P 10DLC registration status of your phone numbers

Please add an emergency address to this phone number or you may incure a $75.00 charge per emergency call. **Add emergency address**

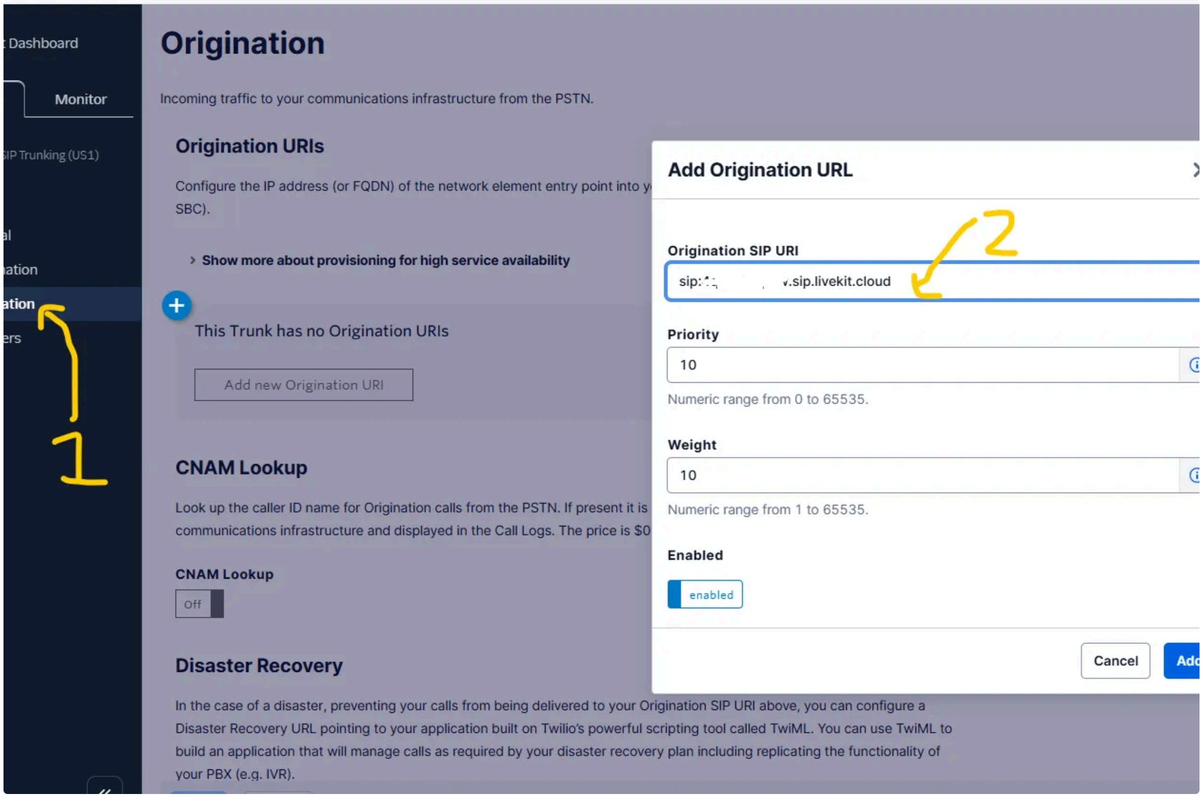
Voice Configuration

Routing: Regional

United States (US1) Region call routing is: **Active**

4. Create Elastic SIP Trunk → Give it a friendly name, i named mine lk_2
5. Go to Origination → Add new Origination URI → paste in your URI from Step 2.

Skip to content



Origination

Incoming traffic to your communications infrastructure from the PSTN.

Origination URIs

Configure the IP address (or FQDN) of the network element entry point into y SBC).

› Show more about provisioning for high service availability

This Trunk has no Origination URIs

Add new Origination URI

CNAM Lookup

Look up the caller ID name for Origination calls from the PSTN. If present it is communications infrastructure and displayed in the Call Logs. The price is $0

CNAM Lookup

Off

Disaster Recovery

In the case of a disaster, preventing your calls from being delivered to your Origination SIP URI above, you can configure a Disaster Recovery URL pointing to your application built on Twilio's powerful scripting tool called TwiML. You can use TwiML to build an application that will manage calls as required by your disaster recovery plan including replicating the functionality of your PBX (e.g. IVR).

Add Origination URL

Origination SIP URI

sip:[redacted].sip.livekit.cloud

Priority

10

Numeric range from 0 to 65535.

Weight

10

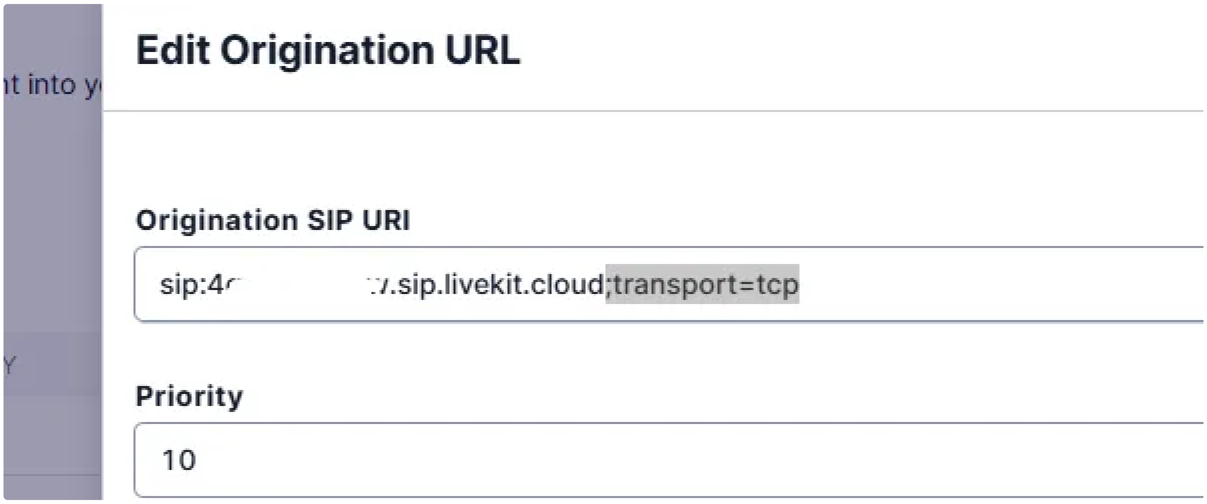
Numeric range from 1 to 65535.

Enabled

enabled

Cancel Add

6. Paste in ;transport=tcp after your URI → Save



Edit Origination URL

Origination SIP URI

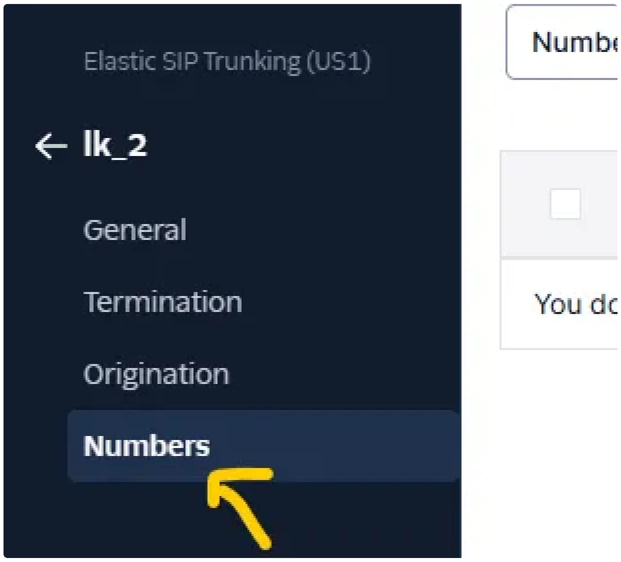
sip:4[redacted].sip.livekit.cloud;transport=tcp

Priority

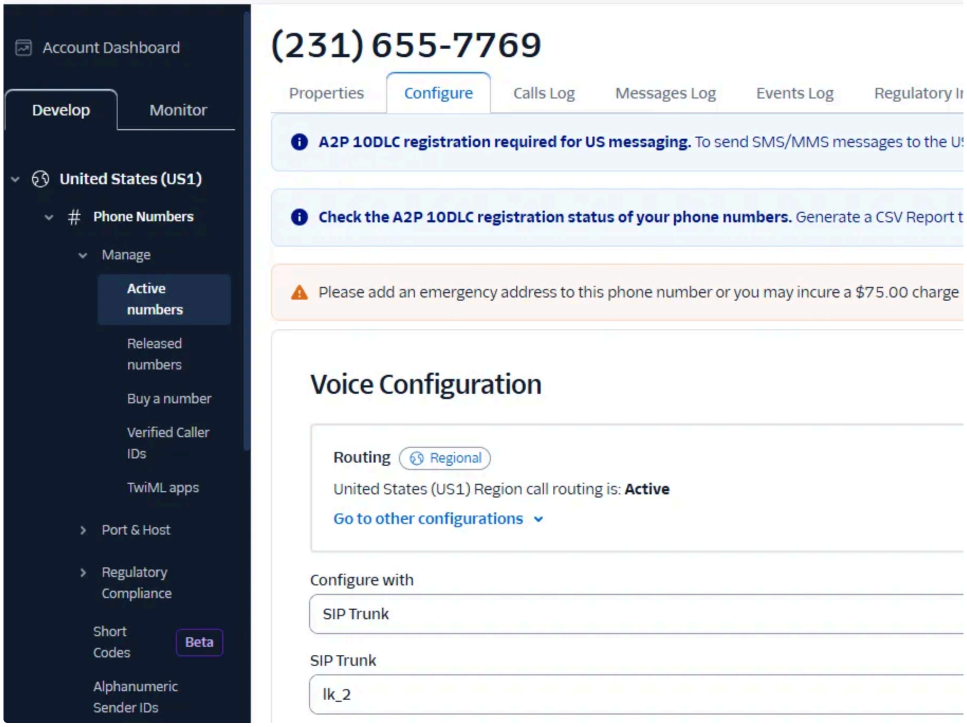
10

Skip to content

7. Click on Numbers and assign your newly bought number to this



And double check in your Active numbers if this number in the Configurations is set with a SIP trunk and assigned to the one you just made



(231) 655-7769

A2P 10DLC registration required for US messaging. To send SMS/MMS messages to the U

Check the A2P 10DLC registration status of your phone numbers. Generate a CSV Report t

Please add an emergency address to this phone number or you may incure a $75.00 charge

Voice Configuration

Routing

United States (US1) Region call routing is: **Active**

Configure with: SIP Trunk

SIP Trunk: lk_2

8. (Optional) Then run each one of these commands one by one to start up a virtual environment, this will is where all the "pip" installs will go (I think of them as python install packages):

   ```
   # Initiate python -m venv venv # Activate venv\Scripts\activate # Turn off your venv deactivate
   ```

   Note: ensure the root folder of your project is being used before running this

9. (Optional) then install the initial packages

   ```
   pip install "livekit-agents[deepgram,openai,cartesia,silero,elevenlabs,turn-detector]~=1.0" "livekit-plugins-noise-cancellation~=0.2" "python-dotenv"
   ```

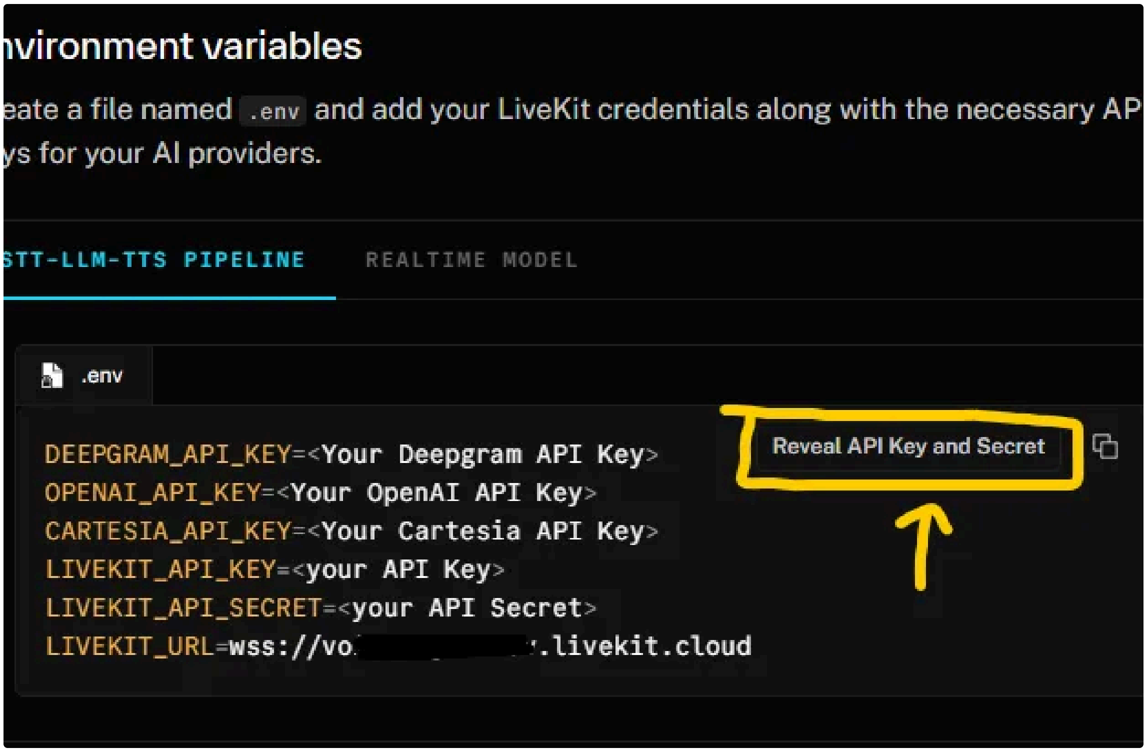
   ```
   pip install "livekit-agents[elevenlabs]~=1.0"
   ```

Skip to content ↪

10. (Optional) create or update a .env with:

```
DEEPGRAM_API_KEY=<Your Deepgram API Key> OPENAI_API_KEY=<Your OpenAI API Key> CARTESIA_API_KEY=<Your Cartesia API Key> LIVEKIT_API_KEY=<your API Key> LIVEKIT_API_SECRET=<your API Secret> LIVEKIT_URL=<LK URL>
```

Note: you can find some of them by clicking on the Reveal API Key and Secret Button once you hover over the .env area in the LiveKit docs: https://docs.livekit.io/agents/start/voice-ai/



nvironment variables

eate a file named .env and add your LiveKit credentials along with the necessary AP ys for your AI providers.

```
DEEPGRAM_API_KEY=<Your Deepgram API Key>
OPENAI_API_KEY=<Your OpenAI API Key>
CARTESIA_API_KEY=<Your Cartesia API Key>
LIVEKIT_API_KEY=<your API Key>
LIVEKIT_API_SECRET=<your API Secret>
LIVEKIT_URL=wss://vo[redacted].livekit.cloud
```

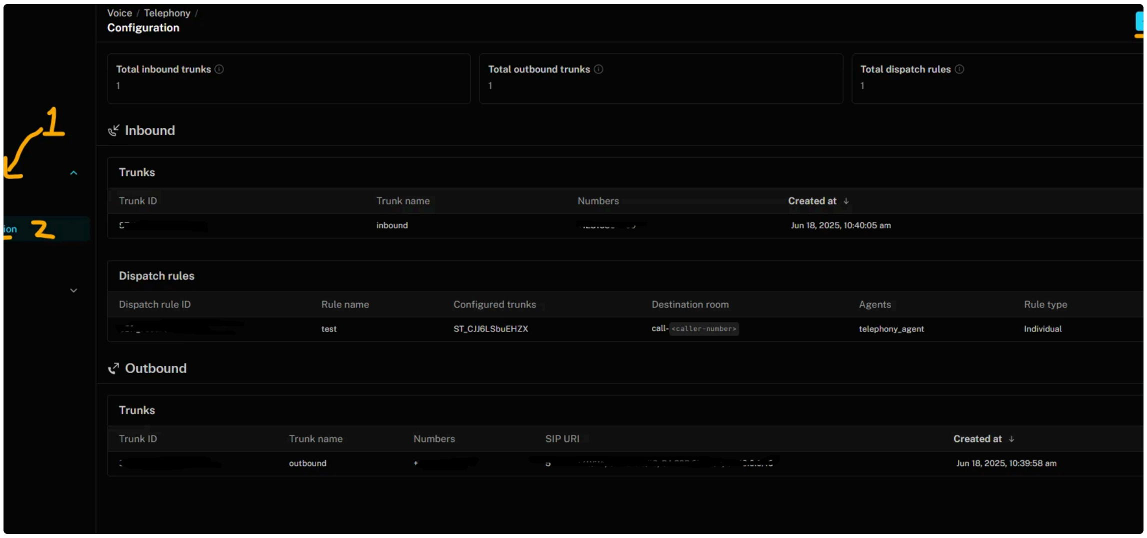
Skip to content

11. Create Inbound SIP trunk in LiveKit

    a. Go to https://cloud.livekit.io and log into your project

    b. Navigate to Telephony → Configuration in the sidebar

    c. Click "Create New" in the top right



Voice / Telephony /
Configuration

Total inbound trunks	Total outbound trunks	Total dispatch rules
1	1	1

Inbound

Trunks

Trunk ID	Trunk name	Numbers	Created at ↓
[redacted]	inbound	[redacted]	Jun 18, 2025, 10:40:05 am

Dispatch rules

Dispatch rule ID	Rule name	Configured trunks	Destination room	Agents	Rule type
[redacted]	test	ST_CJJ6LSbuEHZX	call-<caller-number>	telephony_agent	Individual

Outbound

Trunks

Trunk ID	Trunk name	Numbers	SIP URI	Created at ↓
[redacted]	outbound	[redacted]	[redacted]	Jun 18, 2025, 10:39:58 am

d. Trunk name: "Twilio Trunk" or “inbound” or whatever you want

- **Numbers**: Your Twilio phone number (e.g., +123XXXXXXX)
- **Optional settings**: Leave defaults (authentication is handled automatically)