

Workshop setup: Google's new Generative AI using Angular (30 min)

Author: @gerardsans

Version: 5.0

Prerequisites*

- 1) **Up to date NodeJS** <https://nodejs.org/en/download>
- 2) **Up to date npm:**
`$ npm i -g npm`
- 3) **Up to date Angular CLI:**
`$ npm install -g @angular/cli`

*Optional: VPN access (VPN Express, or similar). Some demos are only available in the US. If you want access to them, install a VPN and select a Region in the US. Eg: [NordVPN monthly plan with 30 days free guarantee](#) (\$12.99)

Mini-Gemini-App

- 1) Make sure you have covered the prerequisites
- 2) Find or create a new folder to work. In the terminal:
`git clone https://github.com/gsans/mini-gemini-app-angular.git`
- 3) Add environment folder and files:
`ng g environments`
- 4) Copy this setup and initialise with your credentials:

```
// src/environments/environment.development.ts
export const environment = {
  API_KEY: "",
  ELEVEN_LABS_API_KEY: "", // optional
  ELEVEN_LABS_VOICE_ID: "21m00Tcm4TlvDq8ikWAM", // optional
};
```
- 5) Create API key using [Google AI Studio](#).
- 6) Install all dependencies:
`npm i --force`
- 7) Run: `ng serve`
- 8) Open browser: `localhost:4200`

Gemini Live App

- 1) Make sure you have covered the prerequisites
- 2) Find or create a new folder to work. In the terminal:

```
git clone https://github.com/gsans/gemini-2-live-angular.git
```

- 3) Add environment folder and files:

```
ng g environments
```

- 4) Copy this setup and initialise with your credentials:

```
// src/environments/environment.development.ts
export const environment = {
  WS_URL:
    "wss://generativelanguage.googleapis.com/ws/google.ai.generativeang
    uage.v1alpha.GenerativeService.BidiGenerateContent",
  API_KEY: "",
};
```

- 5) Optional if you already have one. Create API key using [Google AI Studio](#).
- 6) Install all dependencies:

```
npm i
```

- 7) Run: `ng serve`

- 8) Open browser: `localhost:4200`

Repo quick-test

- 1) Make sure you have covered the prerequisites
- 2) Find or create a new folder to work. In the terminal:

```
git clone
```

```
https://github.com/gsans/gsans-google-ai-gemini-2-angular.git
```

- 3) Add environment folder and files:

```
ng g environments
```

- 4) Copy this setup and initialise with your credentials:

```
// src/environments/environment.development.ts
export const environment = {
  API_KEY: "",
  PROJECT_ID: "", // optional
  GLOUD_AUTH_PRINT_ACCESS_TOKEN: "", // optional
};
```

- 5) Optional if you already have one. Create API key using [Google AI Studio](#).
- 6) Install all dependencies:

```
npm i
```

- 7) Run: `ng serve`

- 8) Open browser: `localhost:4200`

Setup for VertexAI (Google Cloud) - Optional

- 1) Install [Google Cloud CLI](#) (122MB)
 - a) Prerequisite: install Python v3.8-3.10+
<https://www.python.org/downloads/release/python-31011/>
- 2) Sign up for a Google Cloud account and enable billing — this gives you access to Vertex AI.

- 3) Create a new project in the Cloud Console. Make note of the `projectID`.

- 4) Enable the Vertex AI API for your project.

- 5) In the terminal:

```
./gcloud init
```

- 6) Log in to your Google Cloud account (otherwise create one)

- 7) Select projectID with AI enabled.

- 8) Run the command below in the terminal. Make note of the access token

```
gcloud auth print-access-token
```

Setup for ElevenLabs (natural AI voices) - Optional

- 1) Sign In (elevenlabs.io).
- 2) Open the "Profile" menu (bottom left) and select "API Keys" to create your API KEY.
- 3) Add API KEY to the project environment `ELEVEN_LABS_API_KEY`
- 4) Picking a voice `ELEVEN_LABS_VOICE_ID`. You can either use [Voice Lab](#) to listen to the voices, once you pick one, click on "View" to access its "ID" (bottom right) or use the API as shown below:

```
curl -X GET https://api.elevenlabs.io/v1/voices
```

Linktree - resources and slides

https://linktr.ee/confoo_googleai_2025