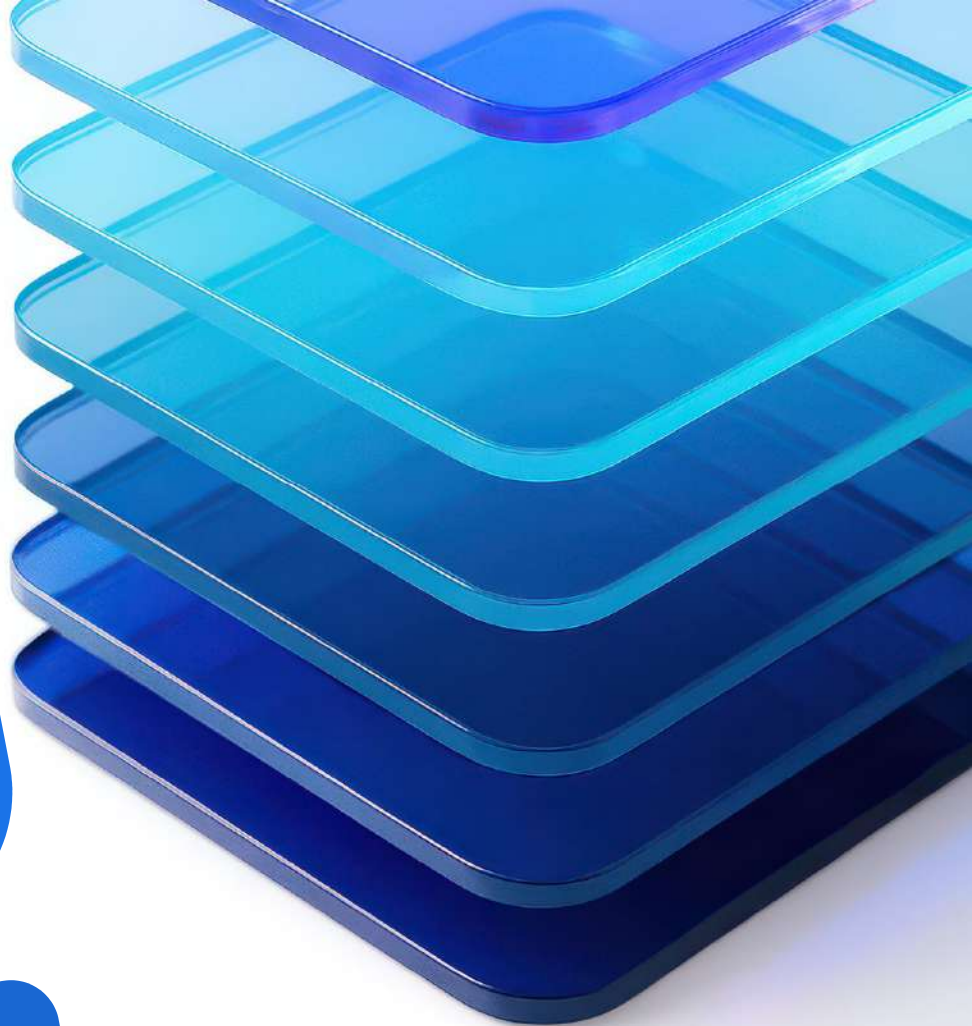


The Voice

*Conversation
vocale live
avec Gemini
en Go*



deleplace@google.com



Valentin Deleplace

Developer Advocate,
Google Cloud





Google Cloud



Gemini



RAD
(DRA)

CLI

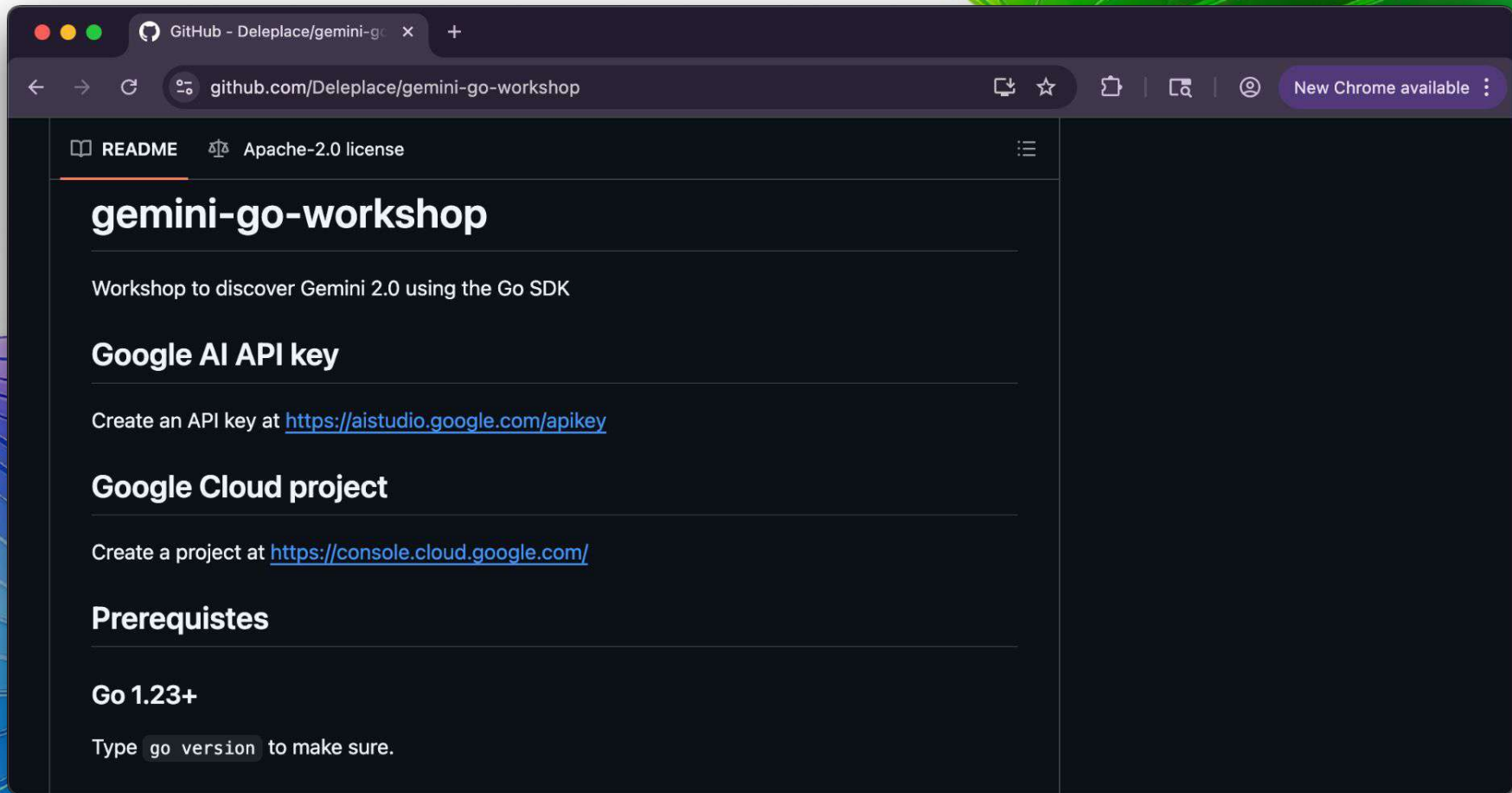


The background features abstract, layered, wavy shapes in shades of blue and green, creating a sense of depth and movement. The blue shapes are on the left, and the green shapes are on the right, both curving towards the center.

voix



ai.dev





GitHub - Deleplace/gemini-go



github.com/Deleplace/gemini-go-workshop

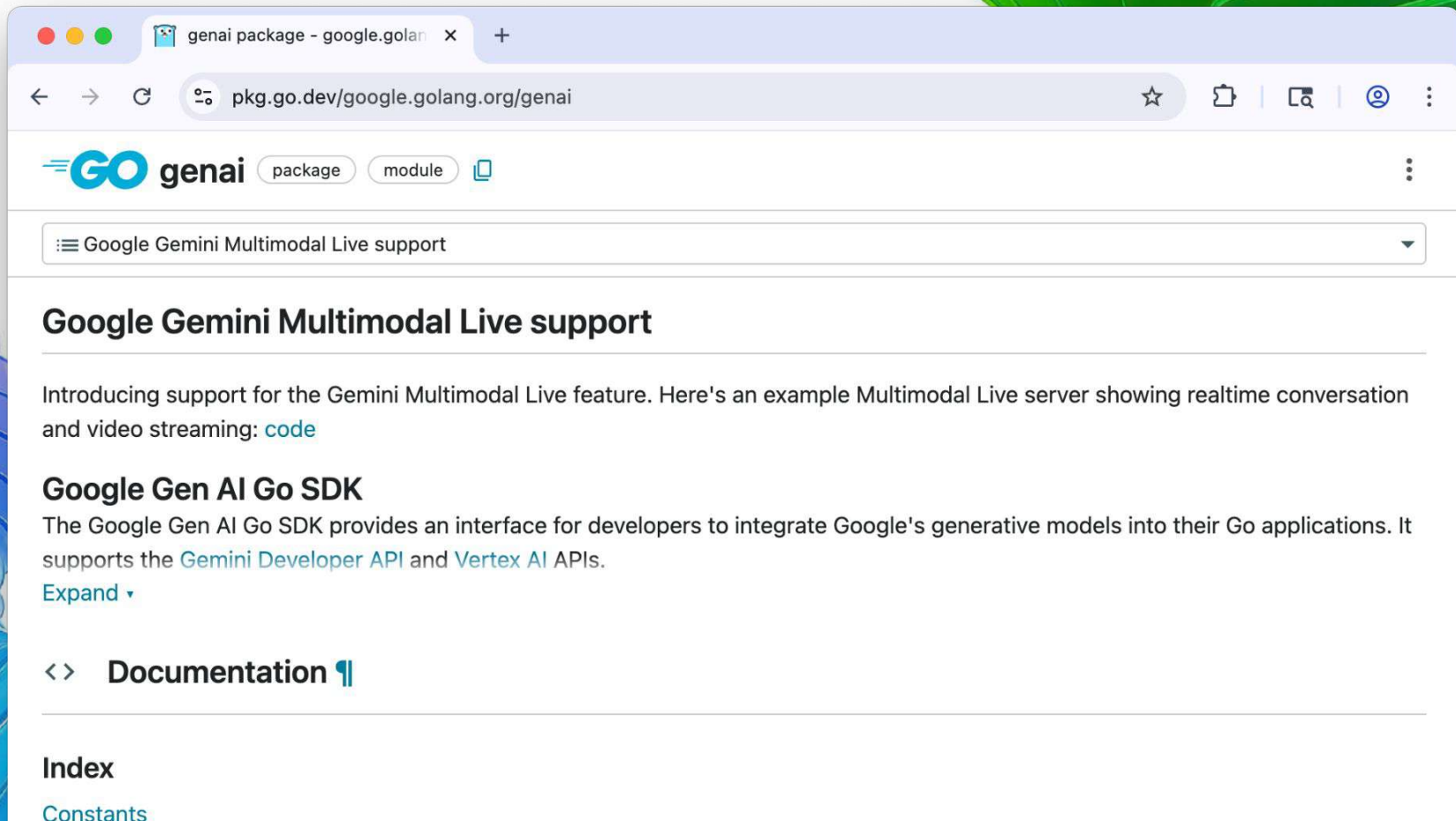


README



Apache-2.0 license

gemini-go-workshop



genai package - google.golang.org

pkg.go.dev/google.golang.org/genai



package

module



Google Gemini Multimodal Live support

Google Gemini Multimodal Live support

Introducing support for the Gemini Multimodal Live feature. Here's an example Multimodal Live server showing realtime conversation and video streaming: [code](#)

Google Gen AI Go SDK

The Google Gen AI Go SDK provides an interface for developers to integrate Google's generative models into their Go applications. It supports the [Gemini Developer API](#) and [Vertex AI](#) APIs.

[Expand](#)

[Documentation](#)

Index

[Constants](#)



genai package - google.golang



pkg.go.dev/google.golang.org/genai



genai

package

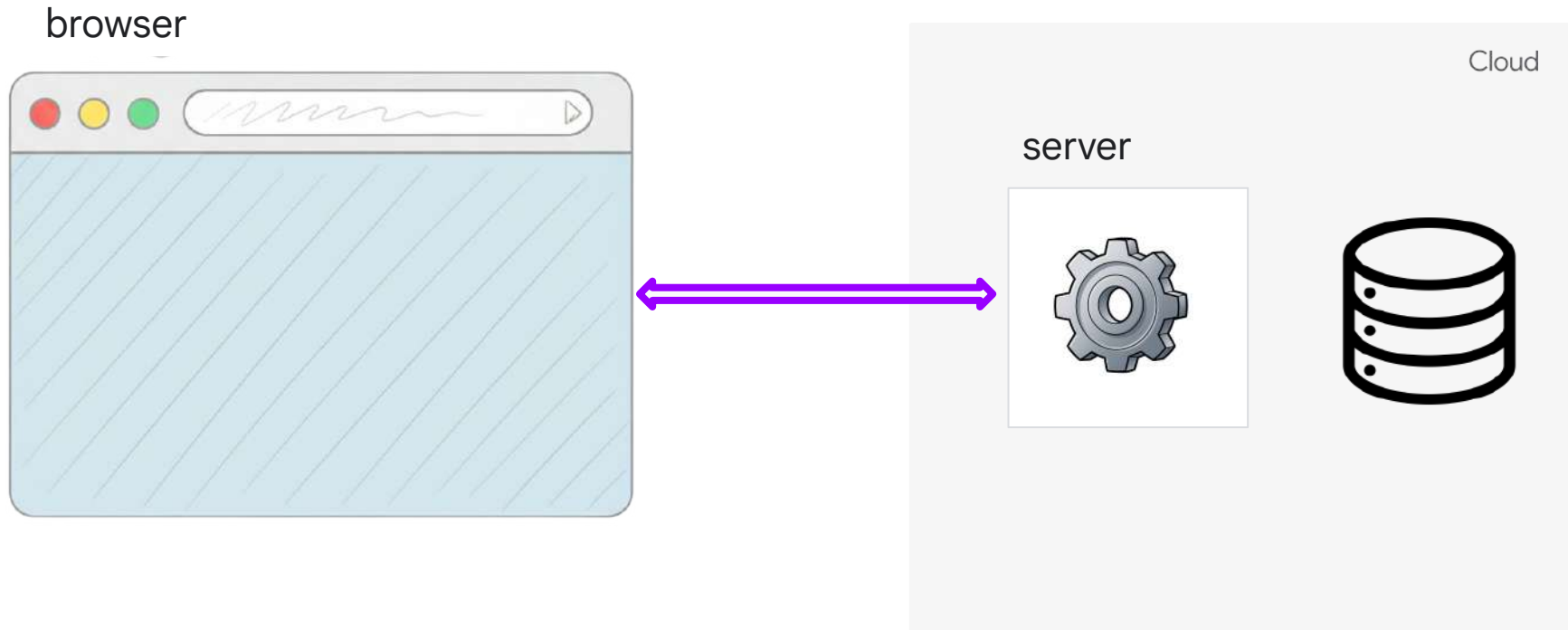
module



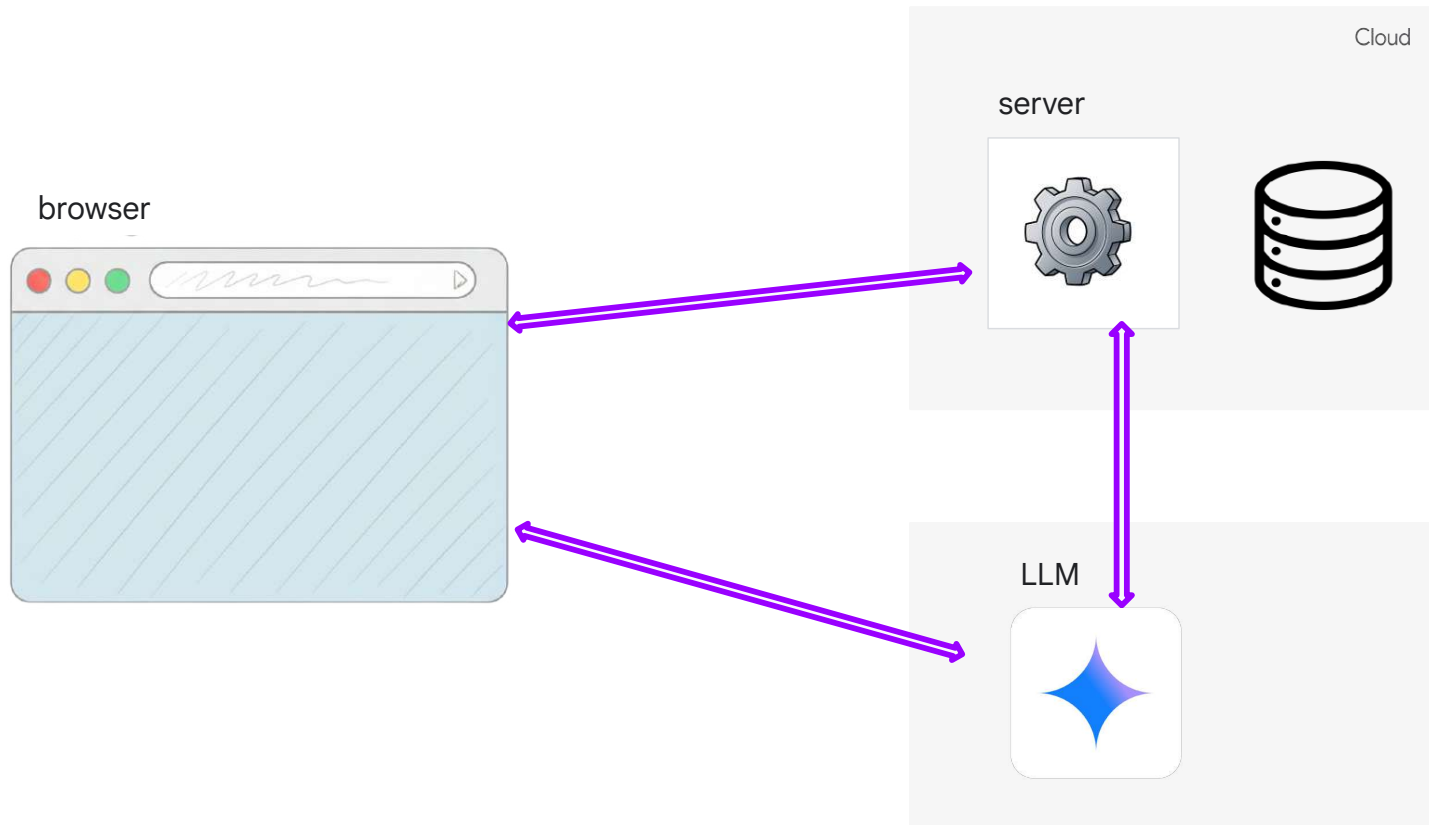
Google Gemini Multimodal Live support

Google Gemini Multimodal Live support

Traditional dilemma



Trilemma



“Local” dev env

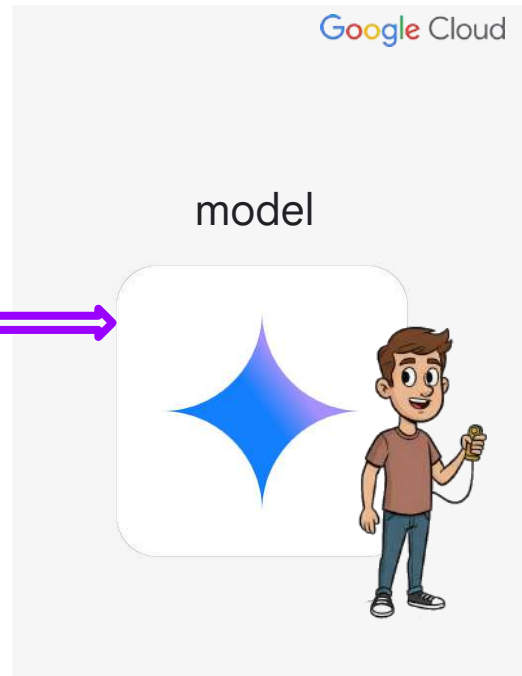
localhost : browser



localhost : server



model



“Local” dev env

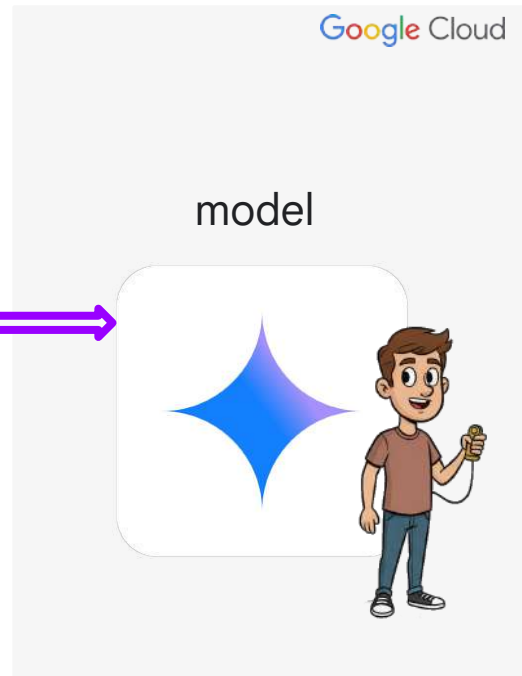
localhost : browser



localhost : server



model



Cloud dev env

browser



Cloud Shell : server

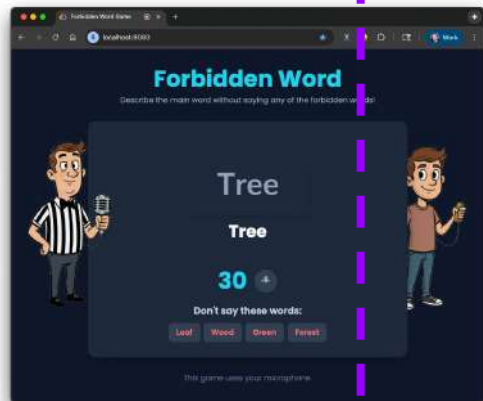


model

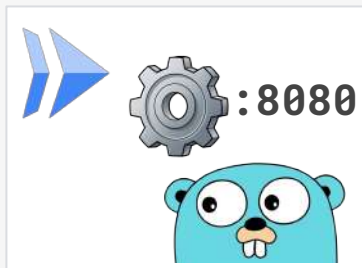


Production env

browser



Cloud Run : server



model



Double live stream?

browser

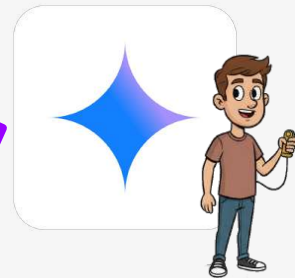


Cloud Shell : server

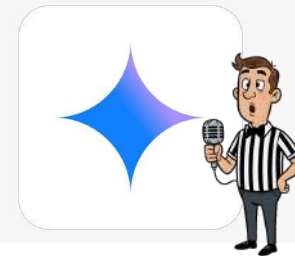


Google Cloud

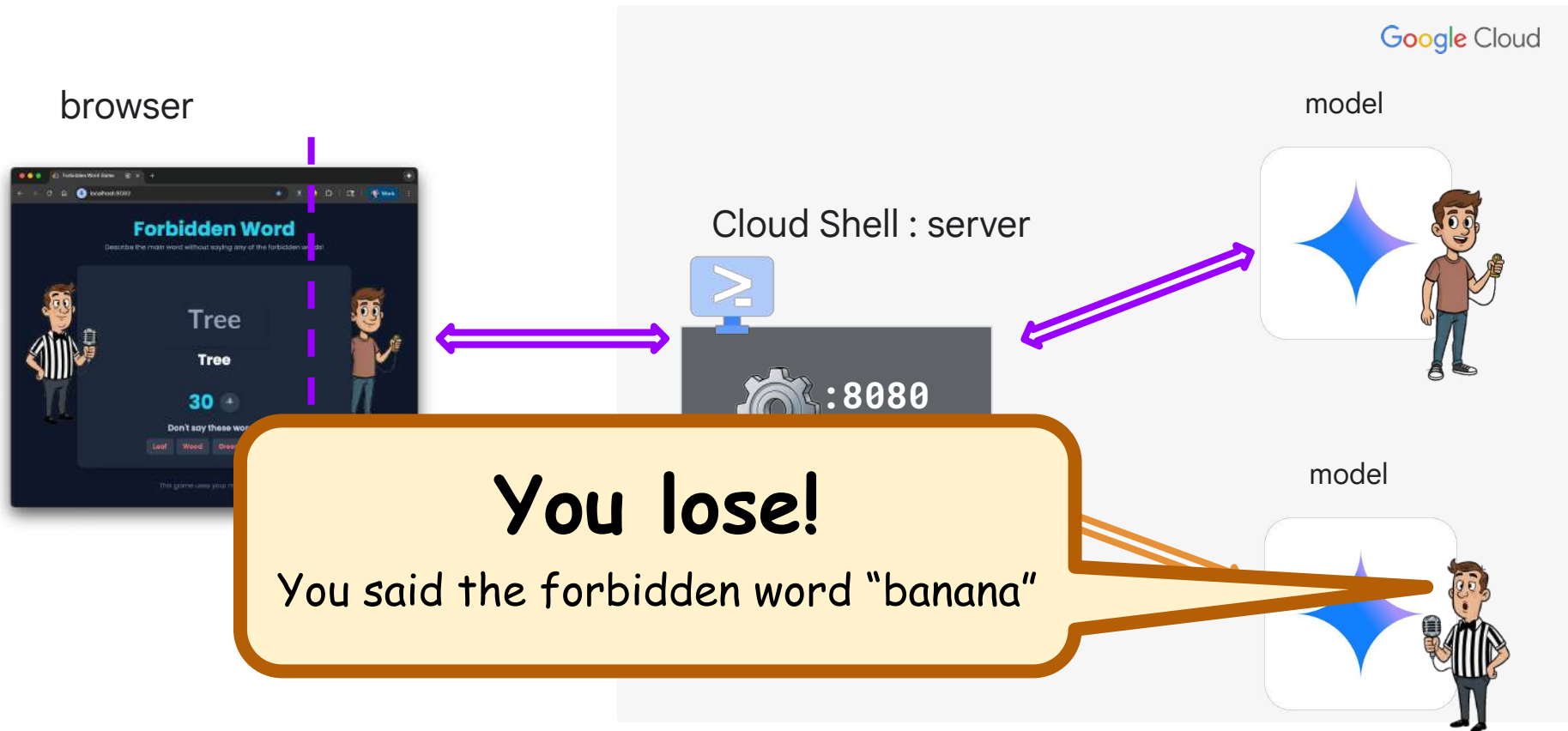
model



model



Double live stream?



Double live stream?

browser

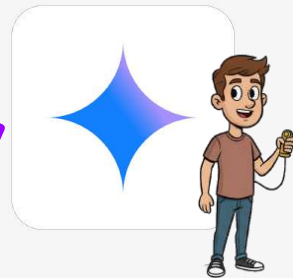


Cloud Shell : server



Google Cloud

model



model



You win

The model guessed "milkshake"

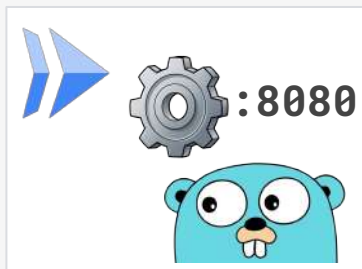
MCP?

Proprietary + Confidential

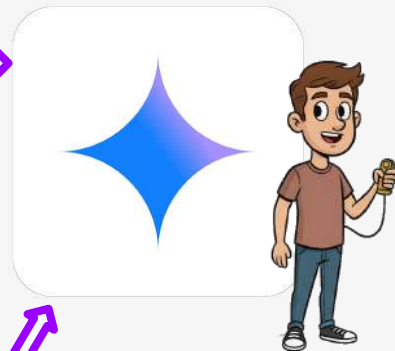
browser



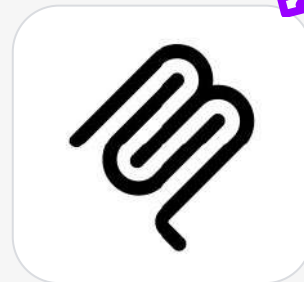
Cloud Run : server



model



MCP



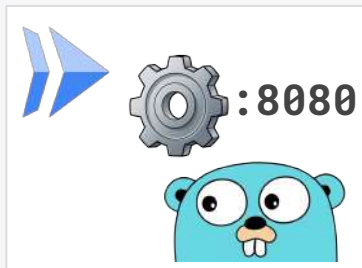
Agent?

Proprietary + Confidential

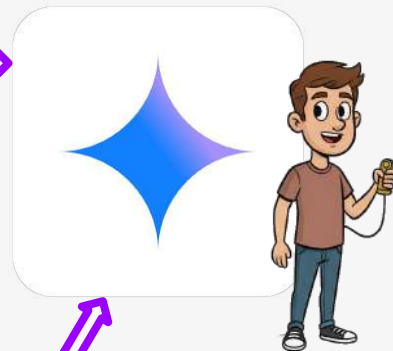
browser



Cloud Run : server



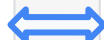
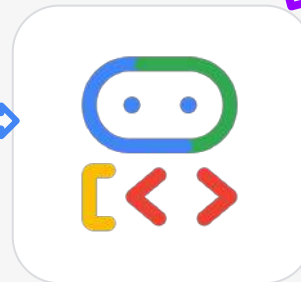
model



MCP



agent



```
package main
```

```
import (  
    "github.com/gorilla/websocket"  
    "google.golang.org/genai"  
)
```



```
config := &genai.LiveConnectConfig{}  
config.SystemInstruction = &genai.Content{
```

```
    Parts: []*genai.Part{
```

```
        {Text: `
```

You are playing the "guessing word" game where the human player with their microphone is describing a word. Your job is to listen to the description and say only one word as your guess, every few seconds. You have only 3 guesses. Don't say anything else than the word you're guessing.

```
        `},
```

```
    },
```

```
}
```

```
voiceName := "Puck"
config.SpeechConfig = &genai.SpeechConfig{
    VoiceConfig: &genai.VoiceConfig{
        PrebuiltVoiceConfig: &genai.PrebuiltVoiceConfig{
            VoiceName: voiceName,
        },
    },
}

config.ResponseModalities = []genai.Modality{genai.ModalityAudio}
config.InputAudioTranscription = &genai.AudioTranscriptionConfig{}
config.OutputAudioTranscription = &genai.AudioTranscriptionConfig{}

session, err := client.Live.Connect(ctx, model, config)
```

```
message, err := session.Receive()

if message.ServerContent != nil {
    if message.ServerContent.InputTranscription != nil {
        inputText := message.ServerContent.InputTranscription.Text
        if inputText != "" {
            log.Printf("Human player says: %s", inputText)
            humanSpeechBuffer = append(humanSpeechBuffer, inputText...)
            log.Printf("Human player says buffer: %s", humanSpeechBuffer)
        }
    }
    if message.ServerContent.OutputTranscription != nil {
        outputText := message.ServerContent.OutputTranscription.Text
        if outputText != "" {
            modelGuessBuffer = append(modelGuessBuffer, outputText...)
            log.Printf("Model player guesses: %s", outputText)
            log.Printf("Model player guesses buffer: %s", modelGuessBuffer)
        }
    }
}
```

```
// Goroutine to receive messages from the GenAI service and send to the client
go func() {
    for {
        // Receive the next message from the GenAI service session.
        message, err := session.Receive()
        //...
        // ...
        err = c.WriteMessage(websocket.TextMessage, message) // Use TextMessage
    }
}
```

Gemini CLI

```
$ gemini -a
```



```
v0.10.0-nightly.20251007.c195a9aa-git.2a7c7166
```

Tips for getting started:

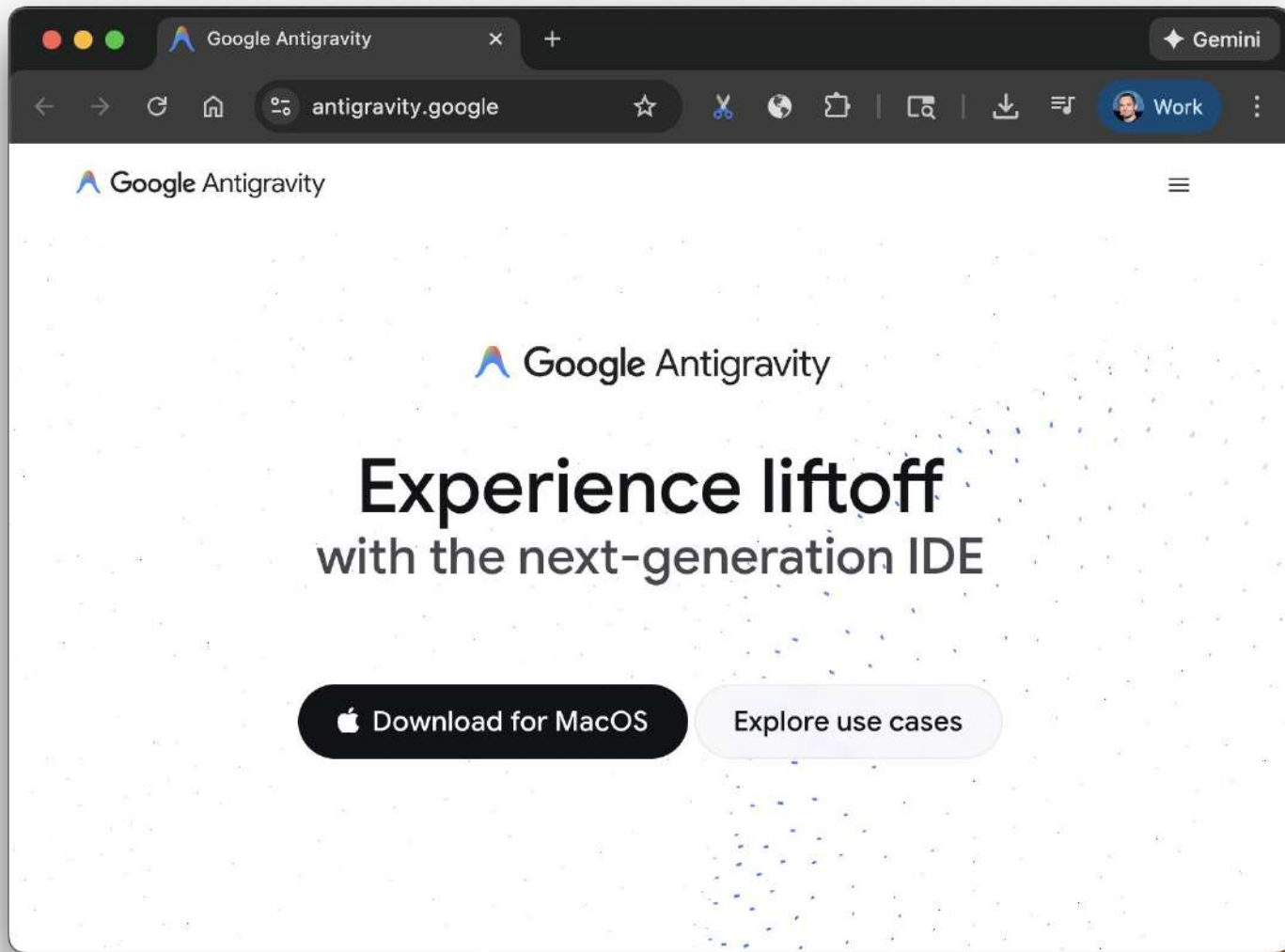
1. Ask questions, edit files, or run commands.
2. Be specific for the best results.
3. Create **GEMINI.md** files to customize your interactions with Gemini.
4. **/help** for more information.

```
> █ Type your message or @path/to/file
```

```
~/Deleplace/gemini-go-workshop (main*)
```

```
no sandbox (see /docs)
```

```
gemini-2.5-pro (100% context left)
```



Google Antigravity Download

antigravity.google/download

☆🔗📄📁📧👤Work

Google Antigravity

ProductUse Cases ▾PricingBlogResources ▾

Download



Download Google Antigravity for MacOS



MacOS

Download for Apple Silicon 

Download for Intel 



Windows

Download for x64 

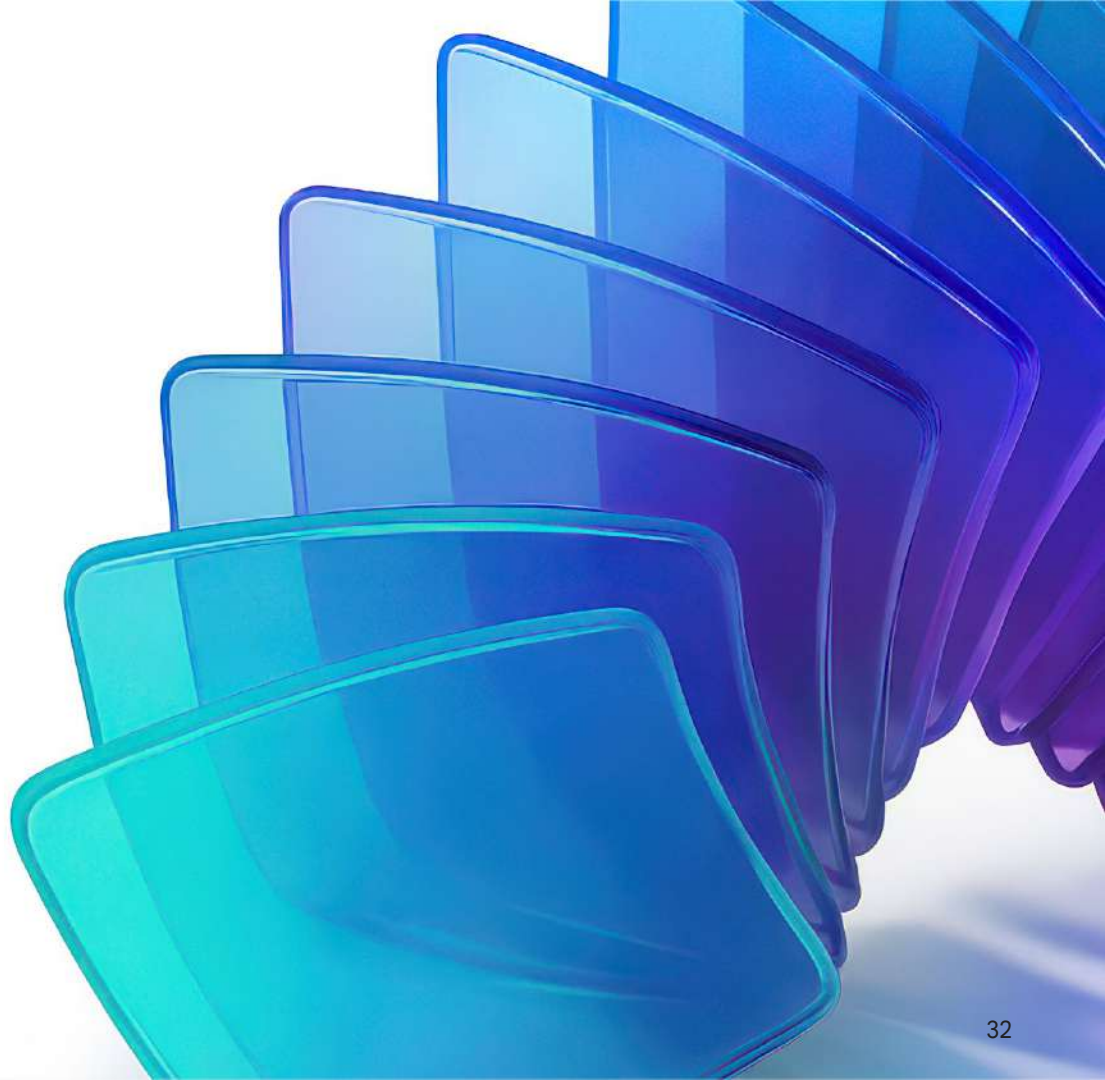
Download for ARM64 



Linux

Download 

Merci :)

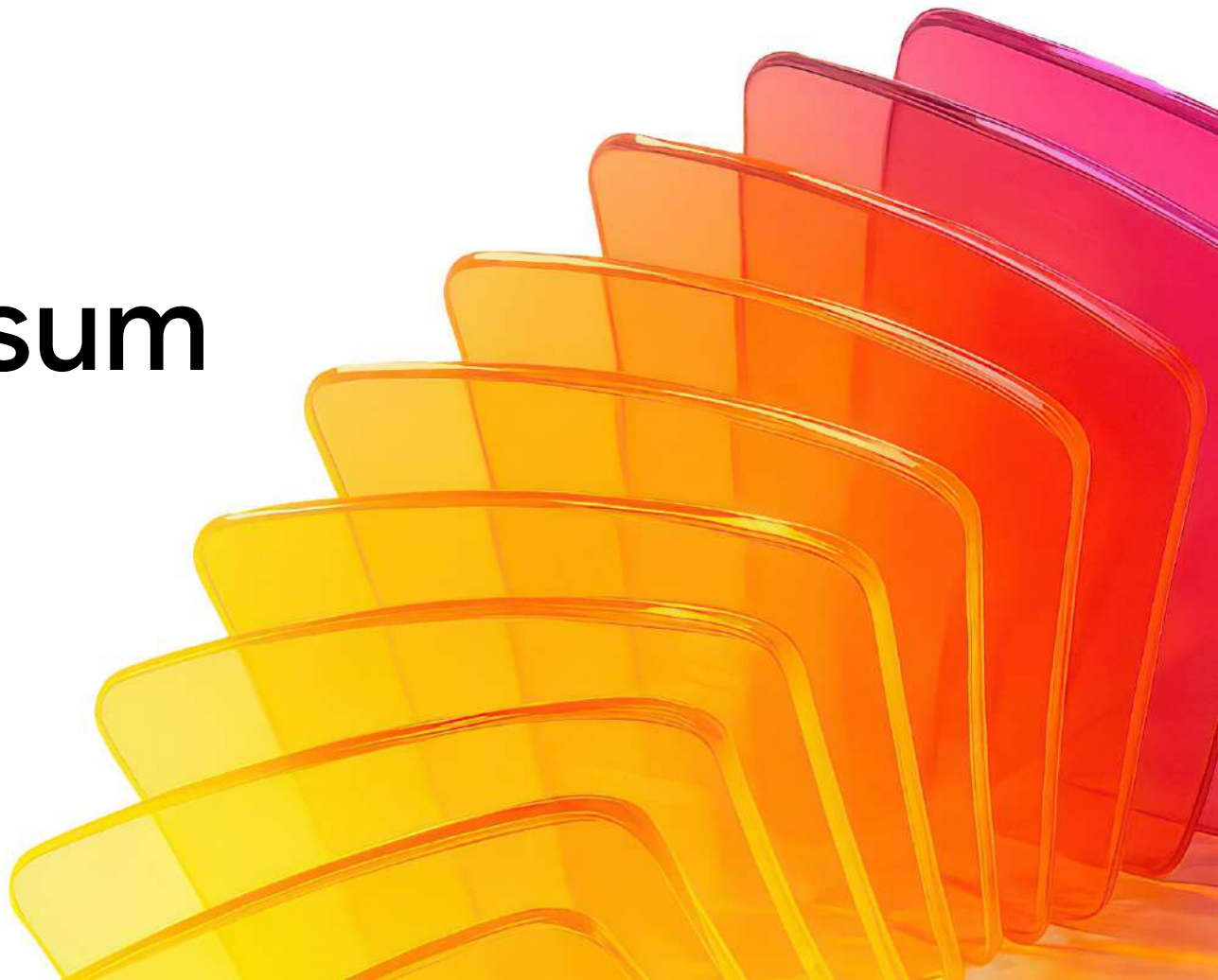


The Voice

Conversation vocale live
avec Gemini en Go



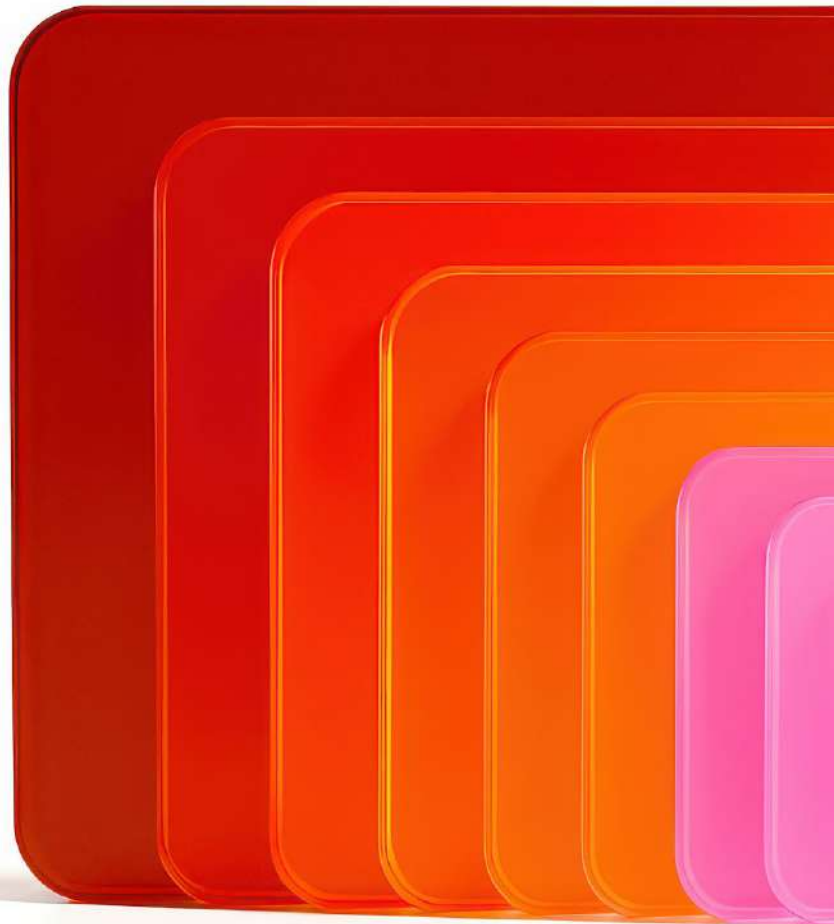
Lorem ipsum dolor sit



Lorem ipsum dolor sit



Lorem ipsum dolor sit



Agenda

12pm Registration + welcome

1pm Opening keynote

2pm Short break

2:20pm Customer panel (blue and red lanyards)

2:50pm

Blue Stage

Unlock Value
with AI Agents
(No Coding Required)

Red Stage

Leverage Agentic AI For
Business Impact:
A strategic workshop

Green Stage

Build your own Agent
with ADK for Developers
(Coding required)

*lanyards are color coded to the breakout sessions

5pm Network Reception

Recommended Speakers

01

15 minutes

Welcome + Opener

Recommended Speaker: Your keynote speaker and/or country director to deliver these opening remarks

02

40-45 minutes

Roundtable Discussions

Recommended Moderators: AI Value Creation leads under monishad@, trained GCC and CE leads, partners

03

15-30 minutes

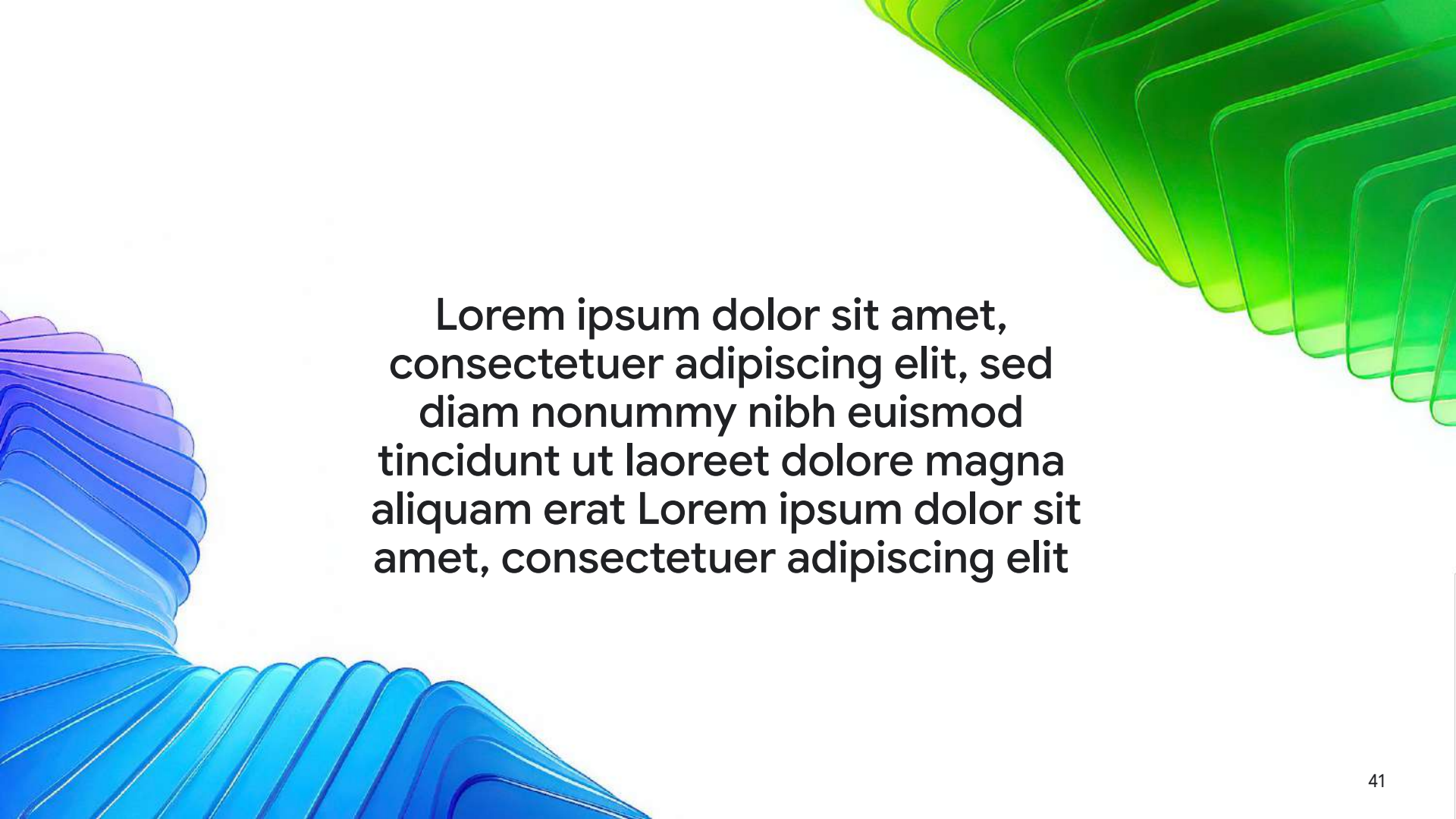
Table read-outs, demo+ agent ideation, and close out the session

(the same person who delivered the opener)

Developer Labs

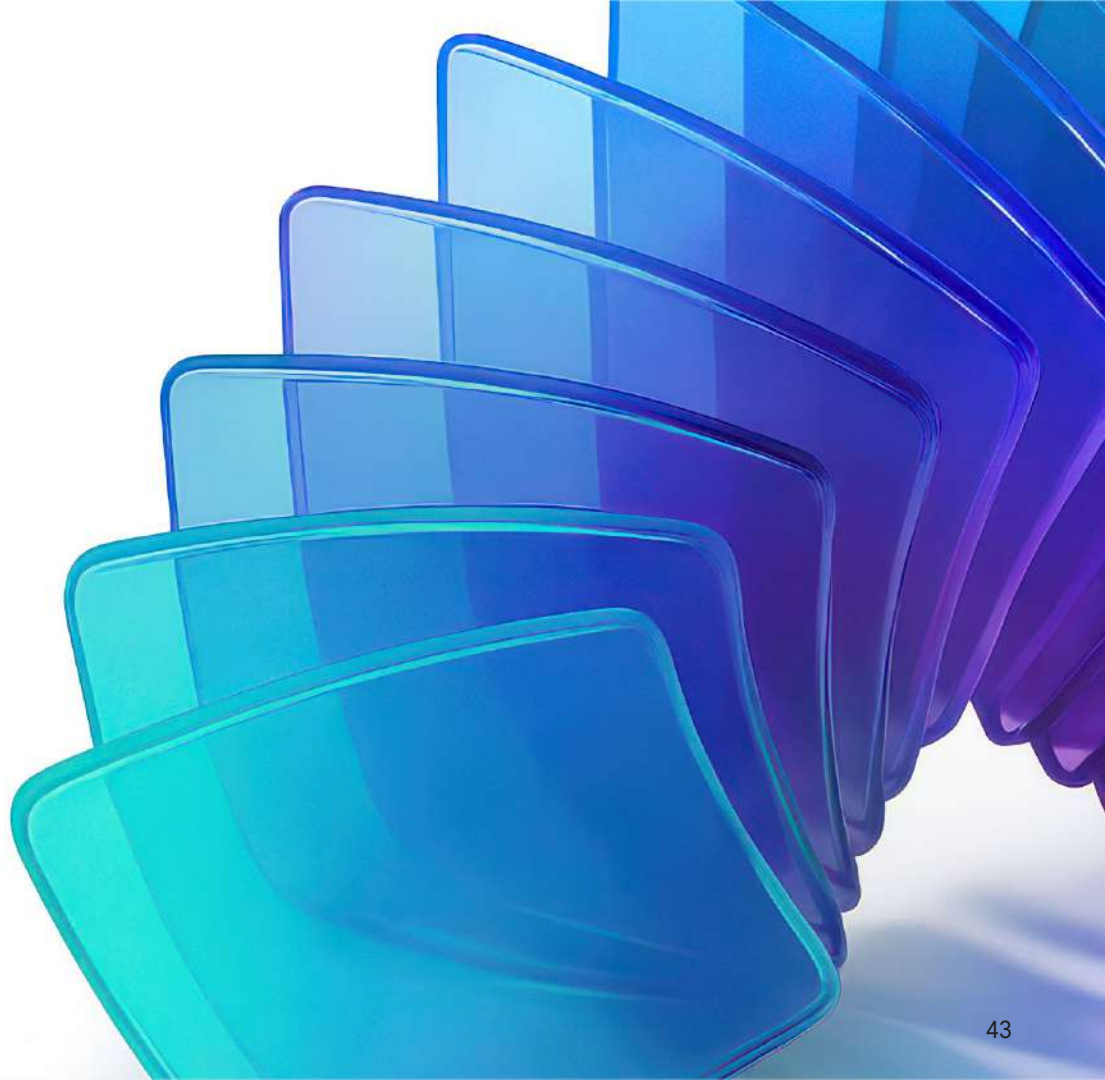
Contact:
Lorem Ipsum

The future is employees + agents

The background features abstract, wavy, layered shapes in shades of blue and green, resembling stylized waves or overlapping pages, framing the central text.

Lorem ipsum dolor sit amet,
consectetuer adipiscing elit, sed
diam nonummy nibh euismod
tincidunt ut laoreet dolore magna
aliquam erat Lorem ipsum dolor sit
amet, consectetur adipiscing elit

Lorem ipsum dolor sit
amet, consectetur
adipiscing elit, sed diam
nonummy nibh euismod
tincidunt ut laoreet
dolore magna aliquam
erat volutpat. Ut wisi enim
ad minim veniam ut.



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam ut.