Google for Startups

Startup technical guide

Generative media

The models, tools, and infrastructure powering the next generation of creative and multimodal experiences.

Introduction

Startups are redefining and building next-generation creative and multimodal applications. But as this space matures, the competitive moat has fundamentally shifted.

It's no longer just about who can generate the best standalone AI image or video generation app. The competitive advantage now lies in building efficient, context-aware agentic workflows that deliver consistent, synchronized media at scale.

Successful builders are moving far beyond the basic prompt box or "generate" button. They're embedding leading generative media models directly into intelligent workflows, they're using novel orchestration tools, and they're building in the cloud. The result? Dynamic systems that retain persistent context and seamlessly support their creative vision as it grows.

As the Future of AI: Perspectives on generative media for startups report shows, founders are already pushing the boundaries of what's possible with AI in the creative space. Now, it's your turn.

This guide provides the technical blueprint for startups looking to use Google DeepMind's generative media models alongside powerful orchestration tools and cloud infrastructure to bring creative applications to market.

Prefer audio?
Listen to the podcast version of this guide.

Listen now

Made with NotebookLM

Table of contents

Generative media on Google Cloud

Startups can draw upon a comprehensive multimodal portfolio spanning text, image, video, speech, and music to bring their visions to life.

For Google Cloud, making AI helpful means anchoring our foundational models in Google DeepMind's world-class, state-of-the-art research. This gives builders the structural flexibility to innovate fast alongside best-in-class cloud security, governance, and data privacy.

By offering enterprise-grade reliability and unique protections like Google's generated output indemnity, we give startups the confidence to securely deploy generative applications into production at scale.

Explore DeepMind models

Modality

Images

Model

Gemini Image

Detail

Also known as Nano Banana, Gemini Image is a family of image generation and editing models that is built on Gemini and uses advanced reasoning to deliver exceptional prompt adherence so you can create and refine high-fidelity visuals conversationally.

Its most impressive capabilities include rendering accurate text directly into images, maintaining strict character and object consistency across different scenes, and grounding visuals with world knowledge and information from Google Search.

A key strength of Gemini Image lies in its ability to render accurate, multilingual text directly within an image, overcoming a common hurdle in AI image generation.

Gemini Image can use up to 14 reference images, combining or editing them while maintaining the reference characteristics.

Startup blueprint: See the workflow for using Nano Banana Pro to fuse character assets seamlessly into new environments without model retraining.

Visit technique library

Startup case study: See how the webcomics app toongether cut latency and unlocked production-grade character and style consistency across multi-panel comic strips using Nano Banana's multi-turn image generation pipeline.

Read the toongether story

Explore Gemini Image

Prompt guide

Modality

Video

Model

Gemini Omni

Detail

Gemini Omni is a breakthrough model built to accept any input modality and create any output, starting with video.

The natively multimodal reasoning and creation model processes text, images, audio, and video to generate high-quality video outputs. It excels at conversational video editing where instructions build on each other sequentially over multiple turns—while maintaining strict context, character consistency, and realistic scene physics. In time, it will also support additional output modalities.

Gemini Omni has an intuitive understanding of physics forces like gravity, kinetic energy, and fluid dynamics; along with history, science, and cultural context.

Explore Gemini Omni

Prompt guide

Modality

Video

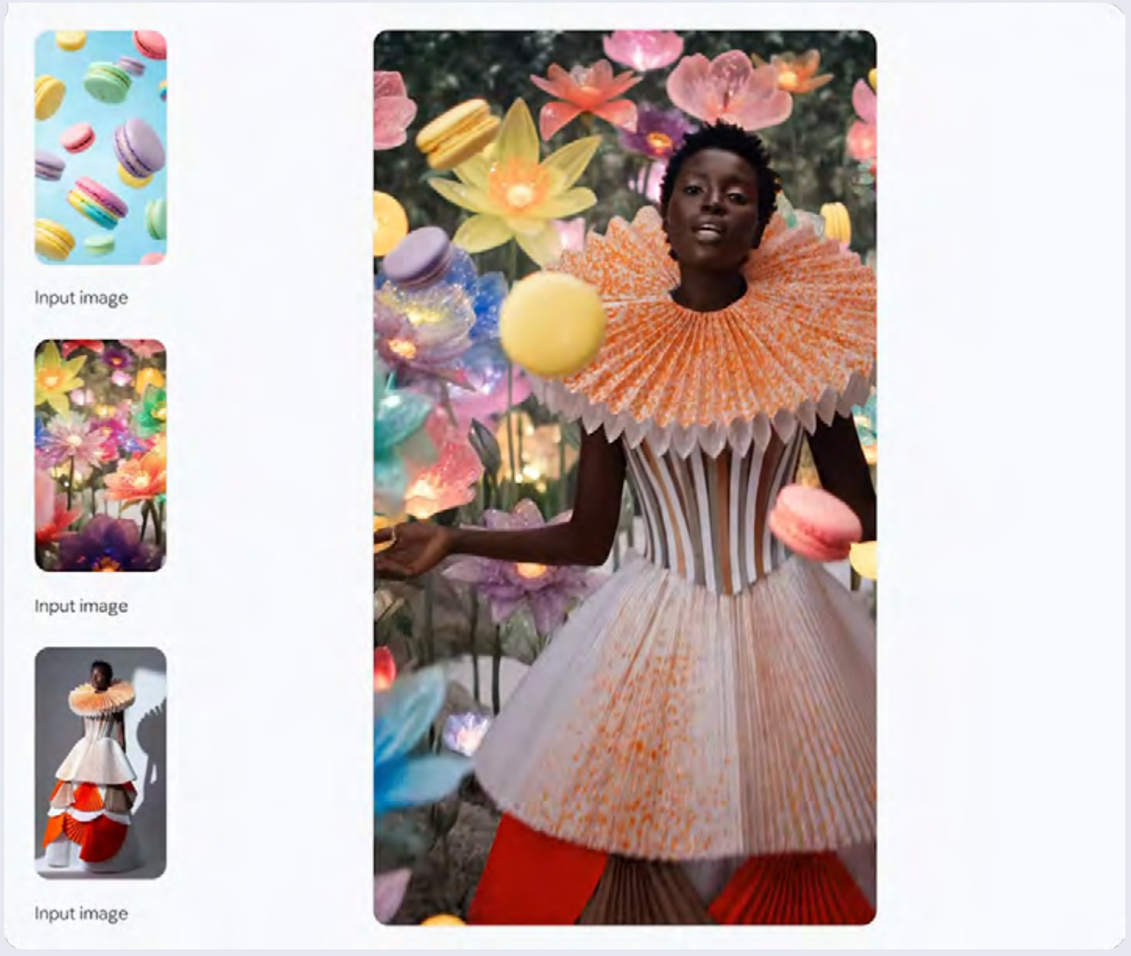
Model

Veo

Detail

Veo is a dedicated video generation model capable of producing high-fidelity 4, 6, or 8-second clips in up to 4K resolution with cinematic realism. It is available at different speeds and costs for various use cases.

It can natively generate synchronized audio—including spoken dialogue, sound effects, and ambient noise—directly alongside the video. It also offers unprecedented creative control, allowing you to use reference images to maintain strict character consistency, and precisely guide transitions by specifying the exact first and last frames of a scene.



With Veo, you can use reference images to maintain character, object, and style consistency.

Startup blueprint: See how Veo is actively chained by an AI agent to turn a raw text script into a fully synchronized sequence.

Visit technique library

Startup case study: See how Synthesia integrates Veo to empower millions of users to generate dynamic, contextually rich video backgrounds for their AI avatars, instantly elevating realism and creative control.

Read the Synthesia story

Explore Veo Prompt guide

Modality

Music

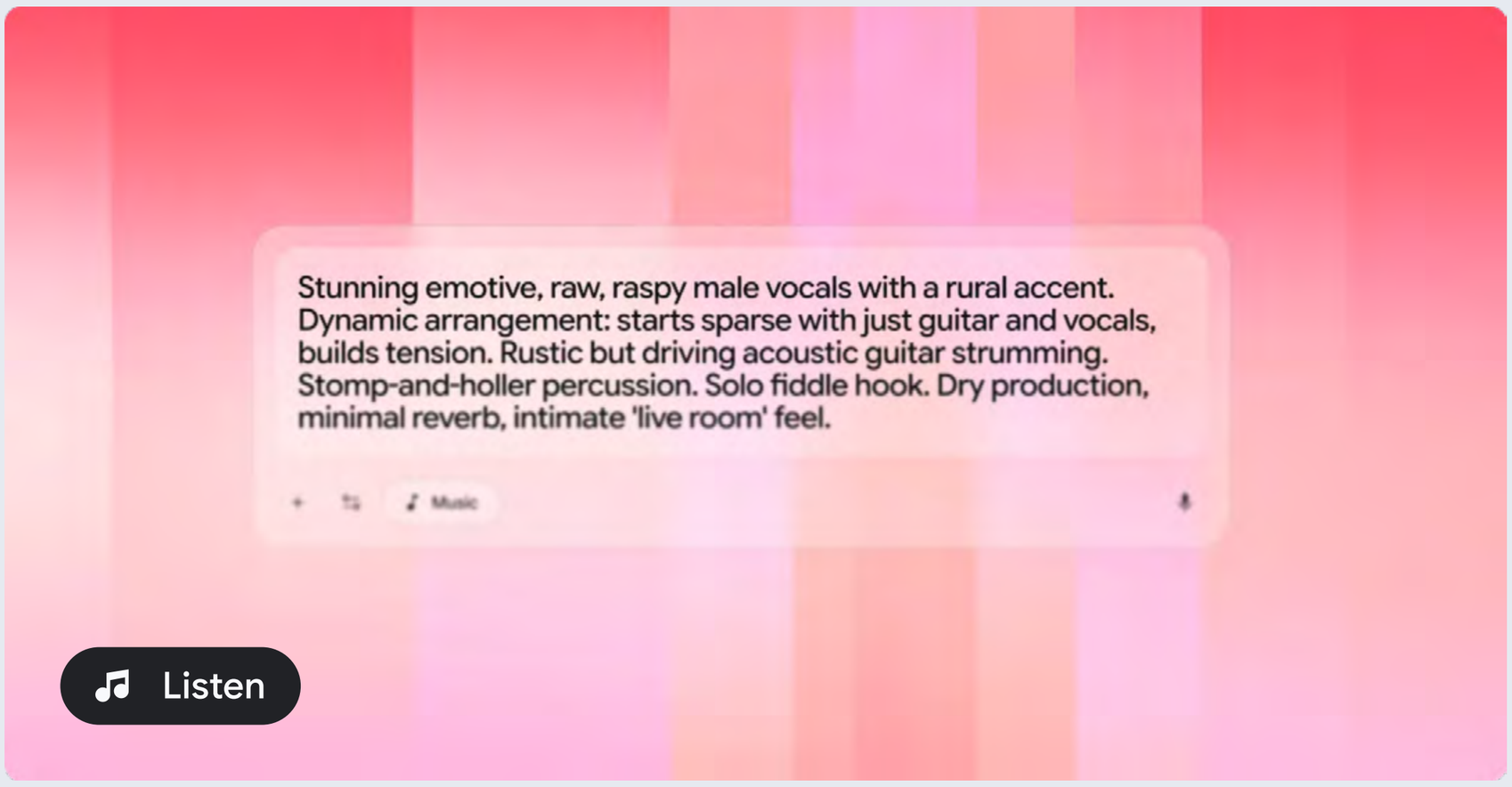
Model

Lyria

Detail

Lyria is a family of music generation models that produce high-fidelity, professional-grade music. These models excel at structural control, allowing you to dictate musical architecture like intros, verses, choruses, and bridges using natural language timestamps.

Lyria can generate vocal performances in eight different languages, giving you the flexibility to guide the overall vocal style and lyrics to perfectly match your composition, and can be used to create 30-second song clips, or full songs up to ~3 minutes long.



With Lyria, conversational prompting allows you to create and control rich soundscapes.

Startup blueprint: See how Lyria is dynamically paired with Gemini's vision endpoint to parse video clips and score them on the fly.

Visit technique library

Startup case study: See how mobile comic platform Toonsutra auto-scores serialized graphic novels at scale and cuts manual production overhead using Lyria to dynamically generate synchronized, culturally resonant background music.

Read the Toonsutra story

Explore Lyria Prompt guide

Modality

Speech

Model

Gemini Audio

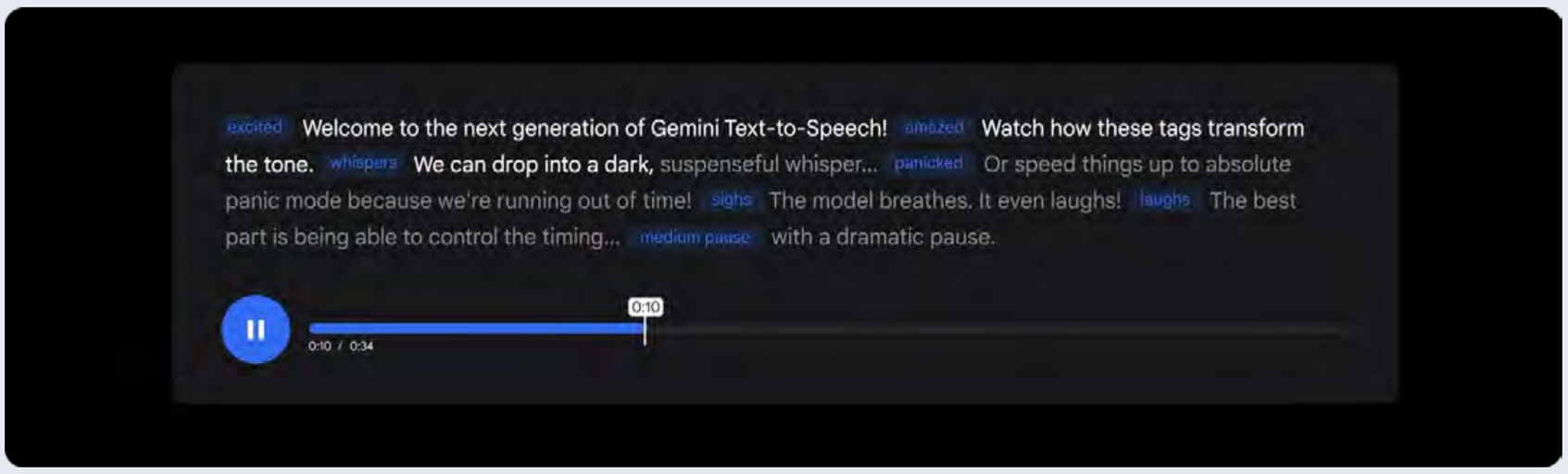
Detail

Gemini Audio is a suite of models designed to process and generate expressive speech.

With Gemini's text-to-speech capabilities (Gemini TTS), you can produce dynamic single or multi-speaker performances by prompting for specific styles, accents, and pacing. It can also translate over 70 languages, using style transfer to preserve the original speaker's unique intonation.

With Gemini Live, you can build low-latency voice and vision agents that interact at the natural speed of conversation. It processes continuous streams of audio, video, and text to deliver fluid spoken dialogue while effectively filtering out background noise. The model provides robust multilingual support, effectively triggers external tools via functions, and strictly adheres to complex instructions to keep your conversational agents securely on track. It also includes a Live Avatar feature.

For audio understanding, it transforms lengthy, unstructured recordings into clean data, complete with accurate transcriptions, speaker separation, and additional advanced transcription capabilities.



With Gemini TTS, you gain precise control over intonation, expression, and pacing for speech generation.

Startup case study: See how the productivity app Doist powers its "Ramble" feature with the Gemini Flash Live API to instantly convert unstructured, stream-of-consciousness voice input into structured, actionable tasks—allowing users to capture ideas at the speed of thought without manual formatting.

Read the Doist story

Explore Gemini Audio

Modality

Text

Coding

Images

Video

Audio

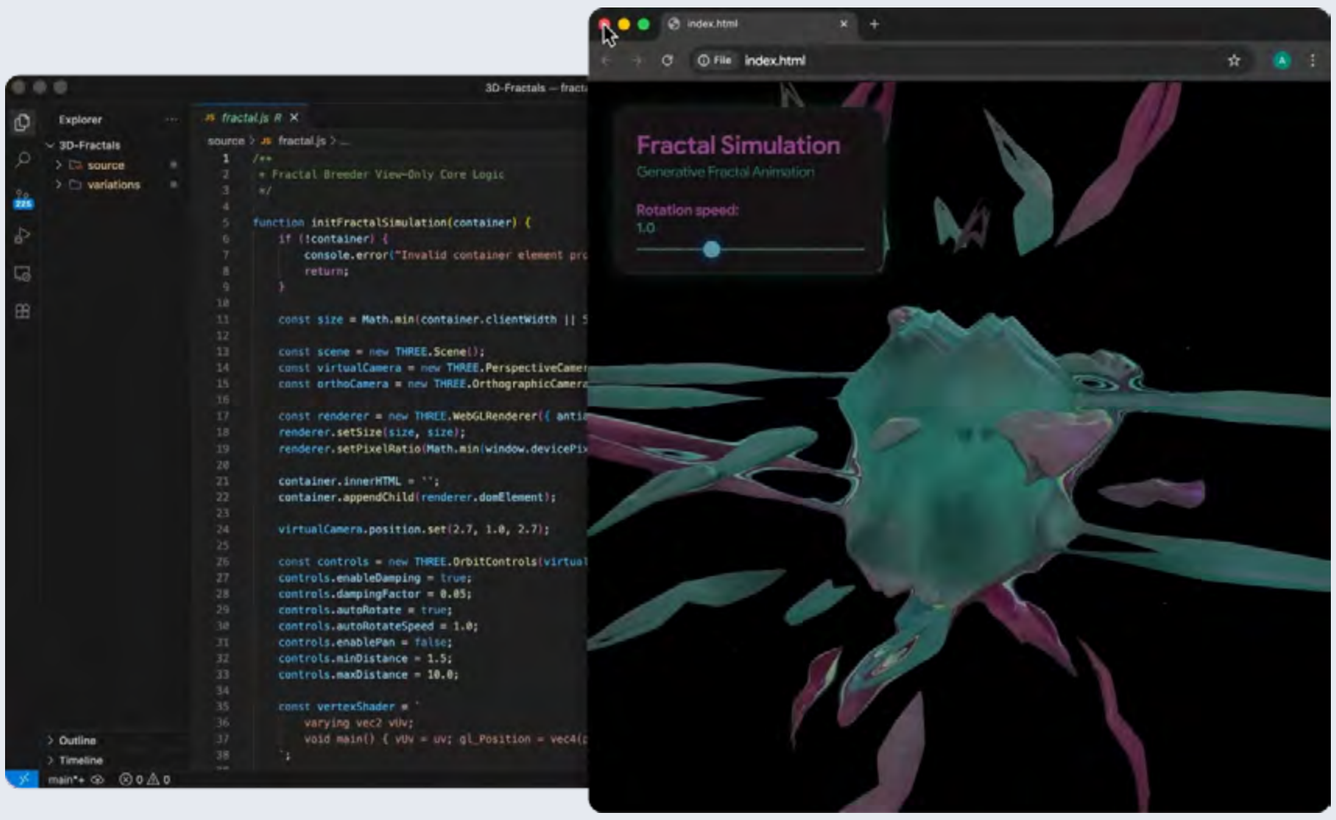
Model

Gemini

Detail

Gemini is a suite of models that possess sophisticated image, video, and audio understanding capabilities, and lead the benchmarks for these tasks.

For generative media agents, Gemini serves as a core engine for advanced reasoning, output quality assurance, agentic coding, and high-velocity multi-step workflows. Gemini models act as the orchestrator for driving downstream creative tools through rapid, parallelized iterative loops.



Gemini excels at advanced coding and deep reasoning, turning complex JavaScript and 3D shaders into live simulations.

Startup case study: See how the employee engagement platform Wotter integrates Gemini's multimodal intelligence to power an AI assistant that analyzes complex, open-ended employee feedback to generate tailored coaching advice and actionable insights for managers, fostering a more transparent and responsive workplace culture.

Watch the Wotter story

Explore Gemini models

Modality

World model

Model

Genie

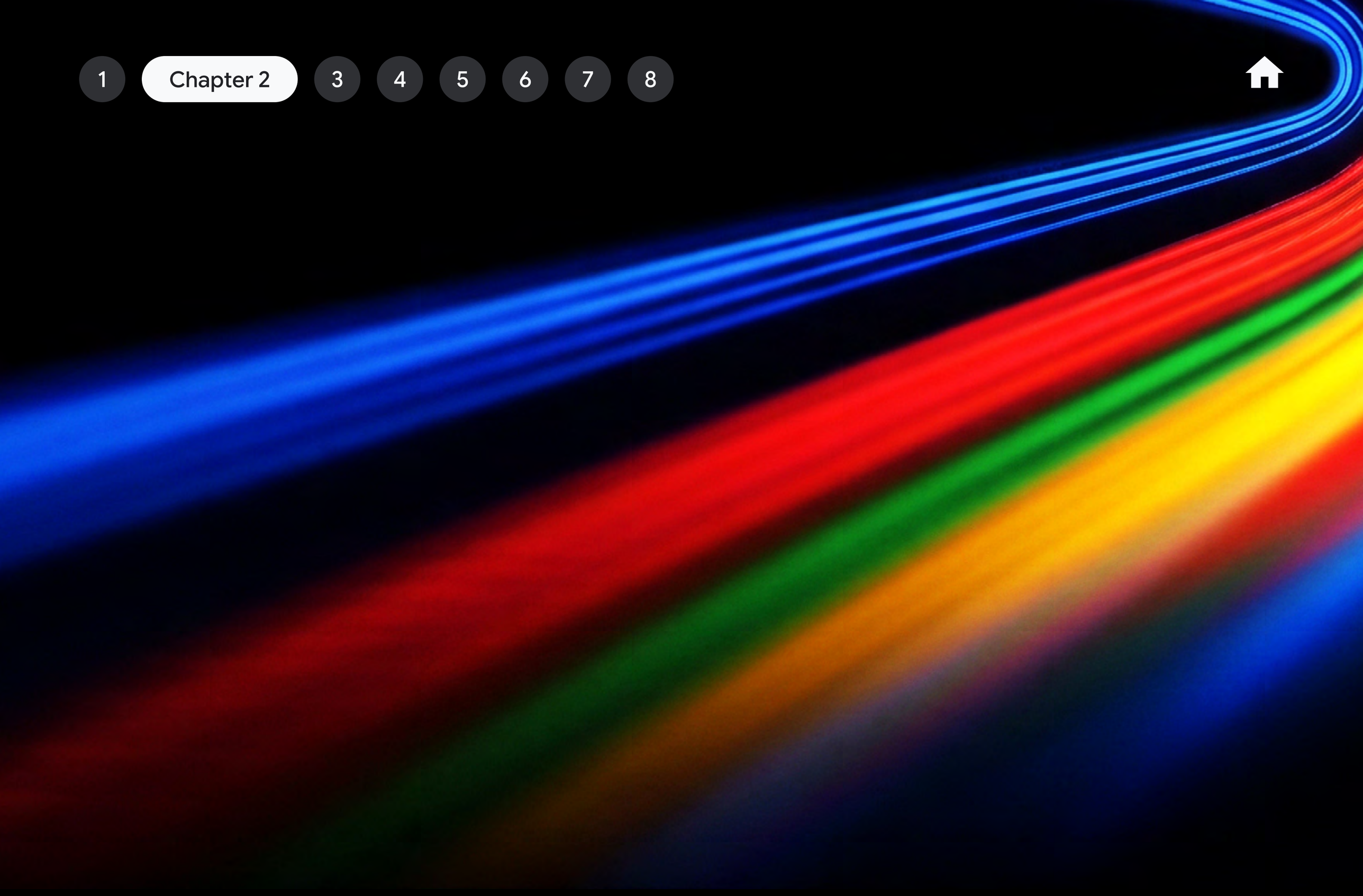
Detail

Genie is a general-purpose world model that uses simple text and image prompts to generate photorealistic environments that can be explored in real-time.

Operating at 20–24 frames per second and 720p resolution, Genie maintains strict environmental consistency and visual stability during extended exploration loops. Grounded in Google Maps Street View data, the model allows users to navigate unexpected scenarios anchored in real-world reality or create complex, physics-grounded spaces from scratch. It serves as a core piece for training generalist embodied AI agents and building next-generation interactive applications.

Genie, which can create photorealistic worlds, is accessible exclusively through our dedicated application.

Explore Genie Prompt guide



Tooling for both prototyping and production

To build a generative media application, you need the right tooling for the creative persona you’re targeting.

Getting started

Startups can begin with rapid, low-friction prototyping. Developers and designers can quickly build interactive UI front-ends with Stitch, design bespoke media tools using Flow, and deploy lightweight applications via Google AI Studio to gain direct, immediate access to Google's complete model portfolio.

Flow

This AI creative studio is designed for generating and editing video and images.

It includes Google Flow Agent, which helps brainstorm, edit, batch-process, and organize media assets; and Google Flow Tools, which allows users to remix and build bespoke creative tools—such as custom shaders or video resizers—using natural language without needing to code.

Explore Flow

Stitch

This AI-native UI/UX design canvas transforms natural language into high-fidelity, interactive UI prototypes. Through "vibe design," users can explore concepts by explaining business objectives.

The platform features an infinite canvas with voice capabilities for real-time critique, and an agent manager to develop multiple ideas in parallel. It also supports DESIGN.md for design system extraction and integrates with developer workflows via the Stitch MCP server.

Explore Stitch

Google AI Studio

This web and mobile based prototyping environment is tailored for rapid experimentation with the Gemini API. Developers can use it to test prompts, adjust safety guardrails, modify model parameters, and ingest multimodal inputs without upfront cost.

The "Get Code" utility eases development by exporting validated prompt configurations into Python, Node.js, or Go snippets using the official `google-genai` SDK. AI Studio also exposes the **Build** tab at ai.dev/apps, a specialized agentic coding environment powered by Antigravity for rapidly prototyping and building with Google models and ecosystem, and sharing, remixing, and deploying AI applications.

Explore Google AI Studio

Advanced orchestration

As your application matures from a simple prototype into a complex, multi-step media pipeline, you need robust developer frameworks and tooling. These platforms provide the granular control, autonomous agent orchestration, and enterprise-grade infrastructure required to build scalable, production-ready systems.

Antigravity

This agent-first development platform, which is designed to orchestrate rather than just write code, features a standalone desktop app that serves as mission control for managing multiple autonomous agents in parallel.

For hands-on coding, Antigravity 2.0 provides a fully featured, AI-powered environment with an agent manager and deep codebase context. Developers preferring terminal interactions can use the Antigravity CLI for high-velocity, headless execution; while the Antigravity SDK provides programmatic access to the underlying agent harness for building custom workflows.

The platform emphasizes transparency by using visual artifacts (like task lists and implementation plans) instead of raw logs to verify agent logic.

Explore Antigravity

Gemini Enterprise Agent Platform

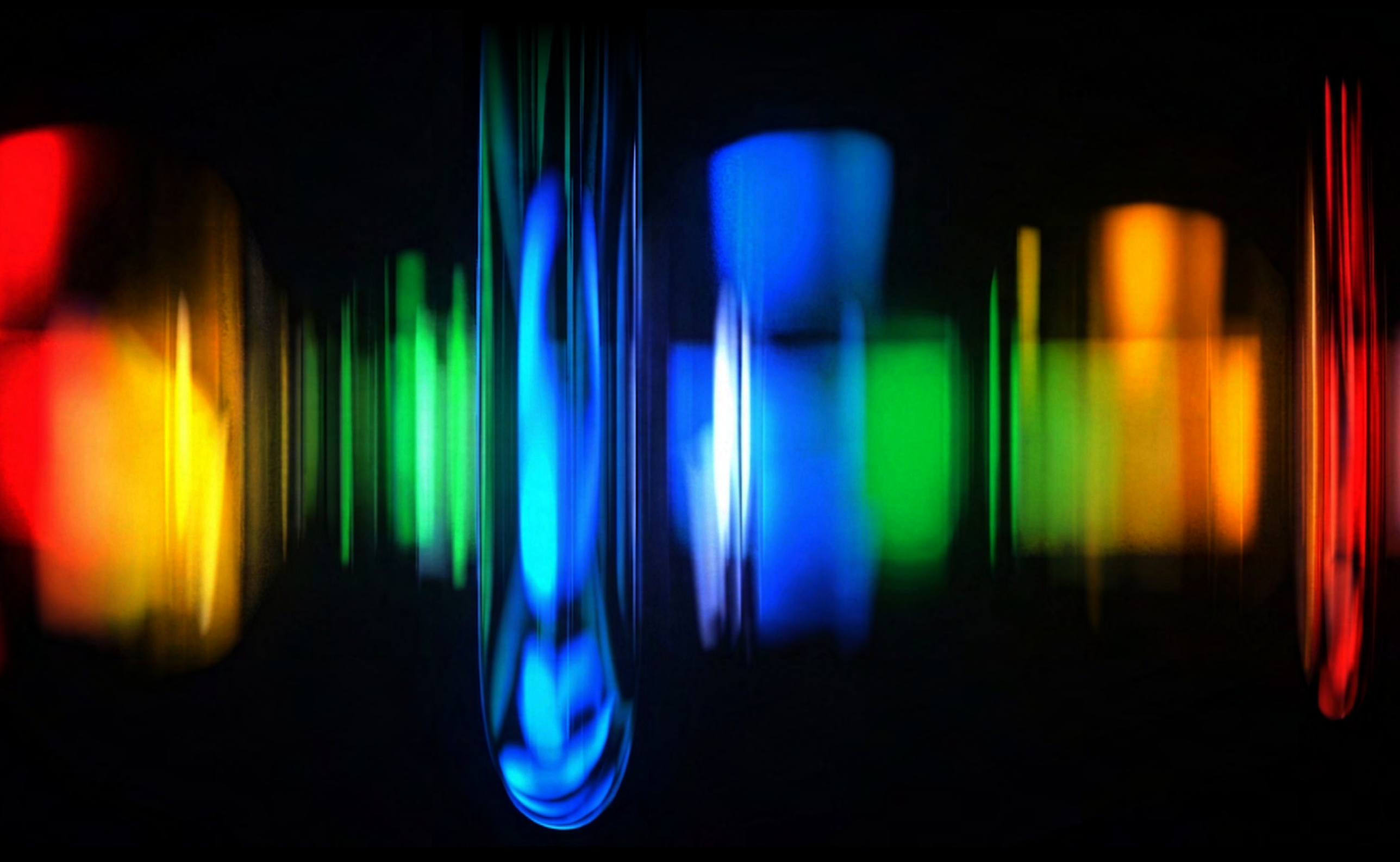
This is Google Cloud's comprehensive platform to build, scale, govern, and optimize AI agents grounded in custom enterprise data.

It acts as a single destination, providing access to over 200 AI models—including the latest Gemini models—as well as third-party and open models. It also includes purpose-built MLOps tools for managing the entire ML lifecycle and the open-source Agent Development Kit (ADK) for fine-grained control of sophisticated agents.

Explore Gemini Enterprise Agent Platform

Startup case study: See how GamerXSociety built their real-time gaming coach, GamerVision, using Google Antigravity to accelerate codebase development and Google Cloud ADK to orchestrate parallel AI agents, enabling instant tactical feedback and automated reward tracking without the constraints of legacy platform APIs.

Read the GamerX story



Prompting for precision and synchronization

To achieve professional, production-grade output from generative models, mechanisms such as programmatic guardrails, deterministic control, and advanced prompting patterns are required.

While consumer applications focus on standalone generative prompts, commercial enterprise applications require models to be embedded within complex, highly governed workflows.

To balance the flexibility of foundation models with the strict consistency demanded by corporate clients, prompting must be automated and data-driven.

What's more, startups must implement pipelines that programmatically compile prompts by pulling from relevant corporate systems.

Depending on the specific workflow, this could include brand style guides, reference assets from Digital Asset Management (DAM) databases, targeting metrics from Customer Data Platforms (CDPs), or structured inventories from BigQuery.

Here are two things to consider when engineering prompts for media assets.

1 Sequential reasoning

By instructing Gemini models to generate prompts specifically for a generative media model like Omni, Veo, or Nana Banana—and describe lighting, camera movement, and character action in coherent sequence—developers can create prompts that achieve far greater adherence to complex instructions.

2 Visual consistency

Maintaining a stable identity is perhaps the biggest challenge in generative media. To ensure a character or brand identity remains stable across different shots or modalities, startups must implement pipelines that programmatically inject reference images and apply style transfer techniques. For example, in order to use a specific logo for a generated asset, it must be added as a reference image for the model being used.

Conversational prompting and editing

Using natural language, individual creators and small, focused teams can build complex, bespoke studio tools for themselves. Autonomous agents make this scale possible, with human oversight at the edge.

Transitioning to an agent-driven model requires a fundamental shift in the developer experience—from how commands are written to how systems are debugged and managed.

Four concepts form the unified framework for building these language-first workflows.

1 Intent as the compiler

In tools like Antigravity, natural language *is* the programming language. Developers describe their intent in plain text (e.g., “Analyze this script for visual cues, anchor the characters using Nano Banana Pro, and animate the scene with Veo”). The system then uses this instruction to automatically construct the underlying execution graph, spawning persona-based specialists to handle the job.

2 Artifact-driven transparency

When deploying autonomous agents, debugging raw logs or complex trace structures becomes unwieldy. Instead of checking terminal logs to see what an agent is doing, human operators can review human-readable artifacts—such as dynamically generated task lists, implementation plans, or visual walkthroughs—to verify, approve, or adjust an agent’s work before heavy rendering begins.

3 The interaction-first feedback loop

Startups can replace static form-filling with bidirectional voice-and-video feedback loops using the Gemini Live API. Creators can narrate changes in real-time (e.g., “Make the lighting more moody”) and the system translates it into executed media updates, bridging the gap between human intuition and machine execution.

4 Markdown knowledge items

To give agents persistent, unstructured context, startups should use Markdown files (like `GEMINI.md`, `AGENTS.md`, or `SKILL.md`) in project directories. These files—which can include the brand style guide, tone of voice guide, or specific creative constraints—guide the agent’s thinking. This way, it becomes a media-aware colleague rather than a blank slate, ensuring that every piece of media aligns with global project parameters.

Selecting a generative media architecture

When designing a generative media application or workflow, choose an architectural pattern that balances engineering velocity with system complexity.

Prompt-to-asset (wrapper pattern)

The simplest pattern builds application wrappers around model APIs. This approach minimizes time-to-market by translating basic user inputs into discrete media files with minimal backend processing.

Custom human-in-the-loop (HITL) workflows

As applications scale, developers must graduate to custom, decoupled architectures to manage compute expenses. A common production pattern isolates low-resolution preview generation from high-definition rendering. For example, Cloud Functions can manage asynchronous model calls to return lightweight drafts for user approval, delaying expensive production rendering until explicitly authorized by the client.

Autonomous agentic architecture

The most advanced tier uses Agent Development Kit (ADK) to build autonomous systems. Instead of following linear pipelines, a developer can program a Director Agent to analyze input assets, construct a multi-shot storyboard, critique its own outputs against a standard style guide, and self-correct generation errors prior to rendering.

Startup case study: See how HeyGen uses Gemini's multimodal intelligence to power its standalone AI Video Agent, automating end-to-end video production—from scriptwriting and asset selection to pacing and cuts—and slashing creation times from days to minutes.

Read the HeyGen story

Using ADK for generative media orchestration

ADK is the core open-source framework for building robust, autonomous agents, helping startups move from linear development to true orchestration.

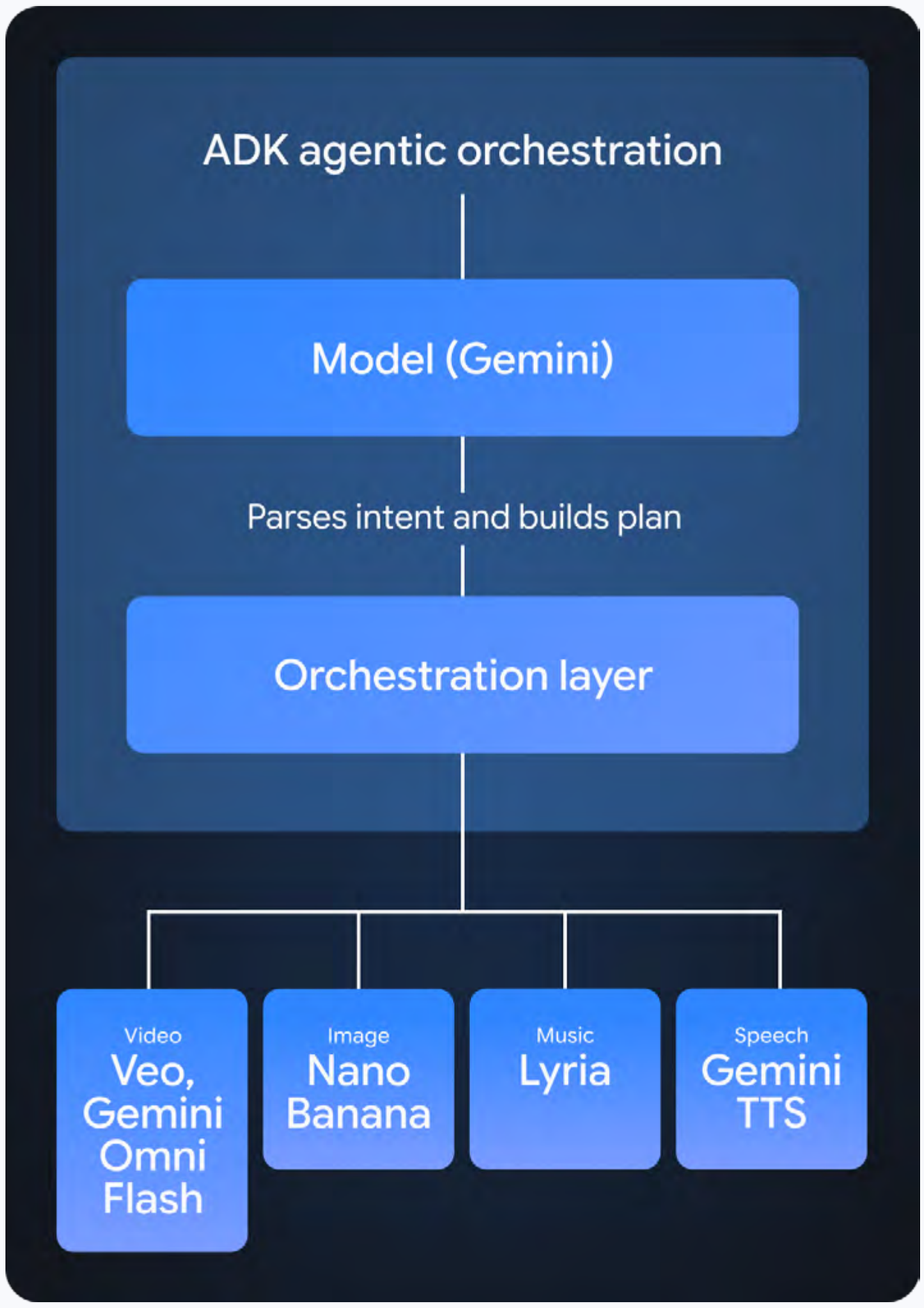
Developers can use it to build agents that possess three essential components: a multimodal model that reasons on the reference content and outputs, tools that take programmatic action, and an orchestration layer for planning and observability.

ADK supports five primary programming languages: Python, TypeScript, Go, Java, and Kotlin. It provides the scaffolding necessary for an agent to solve a complex creative brief, such as a Director Agent scenario.

Pro tip

For developers just getting started, the **Agent Starter Pack** helps accelerate the journey from prototype to production. It includes pre-built templates and handles CI/CD, observability, and infrastructure-as-code best practices.

Learn more



In ADK, the Gemini model acts as the main agent, with specialized generative models acting as the agent's tools, which the ADK-built agent can call and chain together.

Two architectural rules for agentic systems

1 Payload minimization

The orchestration layer can restrict data passing strictly to immutable Google Cloud Storage (GCS) URIs (e.g., gs://bucket_name/asset.mp4). Processing nodes can avoid loading or passing raw binary buffers or image arrays across network boundaries. Instead, they can pass lightweight JSON references, letting downstream model endpoints read directly from the cloud storage bucket layer. This cuts intra-service data transfer overheads and lowers costs when moving heavy media assets.

2 State management via managed session context

To prevent context degradation during extended editing loops, applications must use server-side state tracking. Appended prompt histories, asset seeds, and configuration parameters should be stored in a low-latency cache like Cloud Memorystore (Redis)—passing clean, stateless transaction histories back to the Gemini API rather than attempting to client-serialize model weights or internal attention maps.

Quality assurance via an automated evaluation loop

For a generative media application to become commercially viable, it needs to be reliable. Objective, multi-dimensional evaluation frameworks help catch anomalies like structural clipping or visual warping before they ever reach end users.

Validation is the primary bottleneck in generative media for production ready applications. When a startup scales a platform from producing 10 assets a week to 10,000 localized, personalized variants a day, manual QA becomes impossible.

To guarantee brand safety and visual consistency, platforms must build automated, closed-loop evaluation pipelines, so humans only have to check when truly necessary (e.g., AI ambiguity, arbitration, or adjustment).

Two automated evaluation pipelines can work in tandem to assure quality output.

1 Gemini-as-a-judge agent

This acts as a closed-loop brand guardian or automated auditor. Instead of relying on brittle, code-based heuristics, the reasoning model evaluates the output of the generation models (e.g., Nano Banana, Veo)—looking at the original prompt, the brand guidelines (injected via the `GEMINI.md` context file), and the generated asset. The system semantically verifies if the output adheres to the requested tone, style, and physical constraints.

2 Gecko-based autorater

Google's Gen AI evaluation service uses techniques from Gecko, Google DeepMind's autorater tool, which uses a rubric-based approach to check text-to-image and text-to-video alignment. It moves evaluations for prompt and brand alignment beyond subjective vibe checks by operating through a deterministic, step-by-step breakdown:

1. **Semantic deconstruction**
   A Gemini model parses the initial user prompt into distinct semantic elements —identifying entities, their specific attributes, and spatial relationships.

2. **Rubric generation**
   It dynamically generates a list of strict questions to verify those elements (e.g., "Is the background clear of text?", "Is the car driving on cobblestone?").

3. **Visual probing**
   The autorater visually inspects the generated media against these specific questions, aggregating the results into a granular, interpretable score rather than a vague pass/fail.

Together, these pipelines assure quality

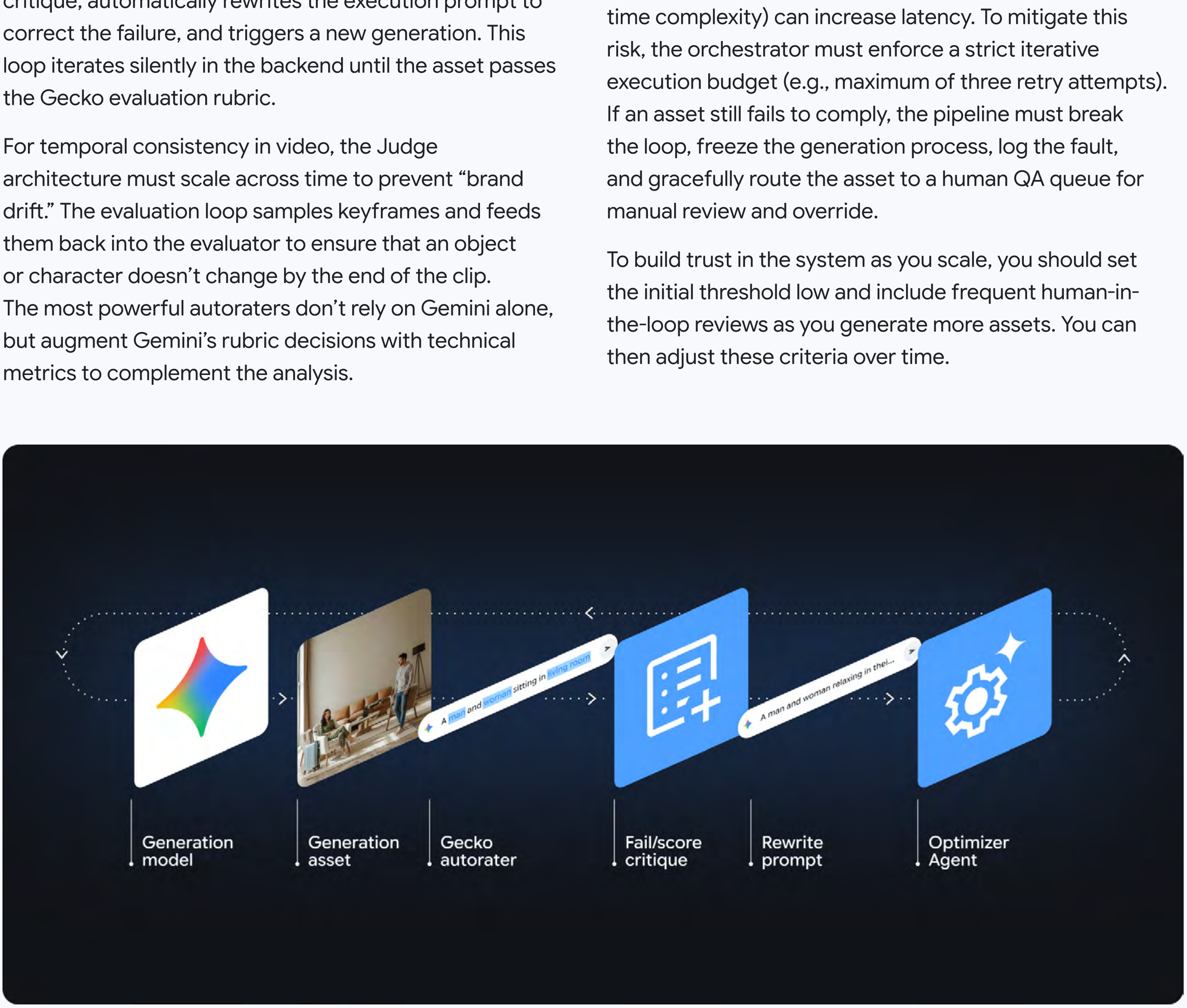
The true power of Gemini-as-a-judge lies in self-correction. When the Gecko autorater flags a composition error (e.g., an object is missing or the lighting doesn't match the mood), the platform doesn't simply throw the error to the user. Instead, the failure log is passed to an Optimizer Agent, which takes the specific critique, automatically rewrites the execution prompt to correct the failure, and triggers a new generation. This loop iterates silently in the backend until the asset passes the Gecko evaluation rubric.

For temporal consistency in video, the Judge architecture must scale across time to prevent "brand drift." The evaluation loop samples keyframes and feeds them back into the evaluator to ensure that an object or character doesn't change by the end of the clip.
The most powerful autoraters don't rely on Gemini alone, but augment Gemini's rubric decisions with technical metrics to complement the analysis.

For images and video, mathematical scoring such as Natural Image Quality Evaluator (NIQE) calculates "naturalness" between scenes, providing a low score for low-blur or lack of artifacts, which can be added to the autorater to ground the assessment in objective metrics.

During self-correction, unbounded recursive loops (O(∞) time complexity) can increase latency. To mitigate this risk, the orchestrator must enforce a strict iterative execution budget (e.g., maximum of three retry attempts). If an asset still fails to comply, the pipeline must break the loop, freeze the generation process, log the fault, and gracefully route the asset to a human QA queue for manual review and override.

To build trust in the system as you scale, you should set the initial threshold low and include frequent human-in-the-loop reviews as you generate more assets. You can then adjust these criteria over time.



Value optimization and economic considerations

As creative technologies become more accessible, individuals and small teams can now produce high-quality media independently. This explosion in content is causing a fundamental shift in economics.

Rather than treating AI simply as a tool to reduce existing production costs, forward-thinking startups are using agentic pipelines to unlock entirely new opportunities and creative capabilities that were technically impossible before.

In traditional content and media frameworks, value was tied directly to production scarcity and fixed capital constraints. In the generative era, the economic moat shifts toward hyper-relevance, scale, and immediate personalization.

The core objective of a production-grade generative media architecture is not merely to render raw pixels efficiently, but to empower creators and enterprises to deliver context-aware, highly engaging experiences that perfectly align with an individual user's real-time needs.

To build a solid economic base for generative media applications, consider these strategies:

Measure Cost Per Interaction (CPI) to quantify scaled relevance

Instead of relying solely on traditional customer acquisition frameworks, evaluate agentic workflows through Cost Per Interaction (CPI). When an automated pipeline handles the complex mechanics of producing thousands of localized, translated, and culturally adjusted asset variants simultaneously, rigid operational barriers are eliminated. Product and marketing teams can execute granular multivariate experimentation at scale, shifting content development from an inflexible upfront capital expenditure into an elastic, data-driven variable asset that adapts to market demand.

Maximize architectural efficiency via context caching

Running advanced multimodal workloads requires a proactive approach to compute management. Context caching is key. If your agent references a massive `GEMINI.md` brand book, a hundred-page script, or a high-resolution library of reference images for every shot, use explicit context caching to ensure that subsequent API calls only incur full costs of the new instructions (the delta), while the cached tokens are billed at a 90% discount. Even with standard storage fees, this drastically reduces overall token costs and latency during high-volume processing.

Implement tiered prototyping and rendering workflows

To protect margins during the iterative phase, applications must divorce the feedback loop from the final render pipeline. A robust architecture uses a Preview/Render split. During the human-in-the-loop phase, the platform generates rapid, low-cost, low-resolution thumbnails (or uses smaller, highly efficient models like Gemini Flash) to lock in the storyboard and composition. Only when the user or Gemini-as-a-judge agent explicitly approves the sequence does the system trigger the computationally expensive, native 4K render via Veo or Nano Banana Pro.

Phase	Model tier	Output target	Compute cost
Iterative drafting loop	**Lean/agile models**	Low-resolution, low-latency thumbnails	Low-latency, optimized token profile
Production export tier	**High-fidelity media models**	Native 4K master assets	High compute depth (isolated behind authorization gateway)

How to avoid synchronization issues

Decoupling the drafting layer from production rendering introduces potential state synchronization vulnerabilities if mutations occur concurrently during final export. To avoid this, the system must enforce atomic state transactions—which you can do by implementing distributed memory locking via Cloud Memorystore (Redis) paired with an event-driven queue using Cloud Tasks. The architecture must explicitly lock and verify the asset state configuration profile before dispatching any final production rendering API call.

Trust, governance, and the synthetic asset lifecycle

Before commercializing your generative media application, you must establish a robust framework that covers both synthetic asset governance and enterprise-grade security.

Global compliance standards keep evolving, as do the security demands of enterprises. Building a foundation of strict governance is a critical competitive advantage that makes your platform market-ready and secure for adoption.

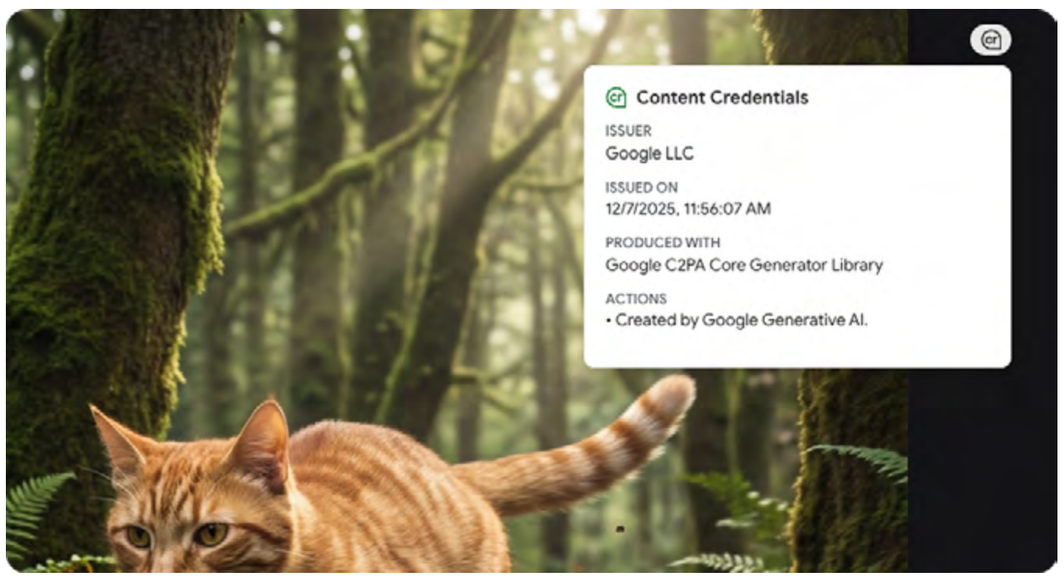
Here's how Google Cloud helps you build this foundation:

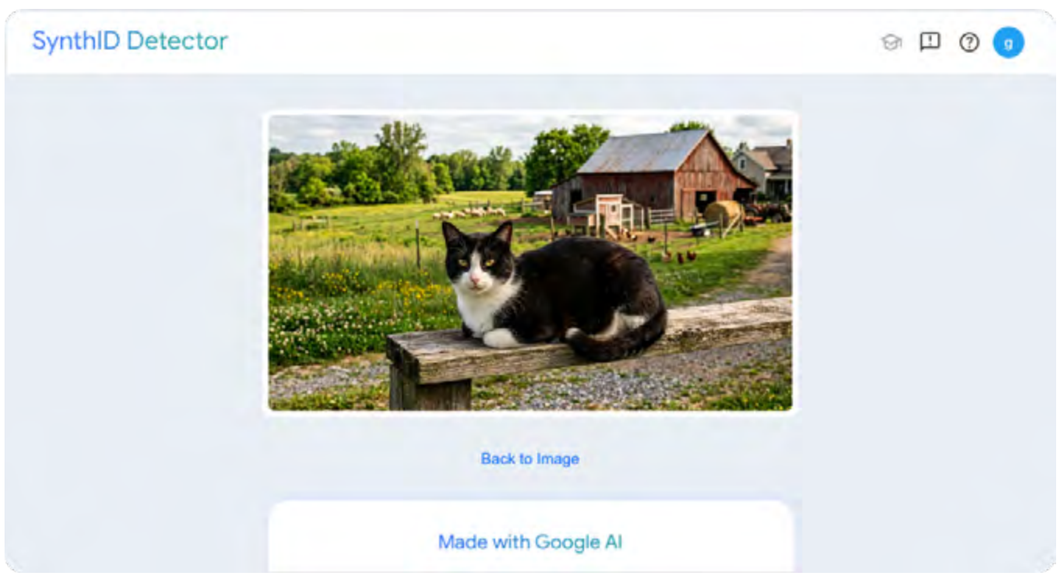
Watermarking with SynthID and C2PA

If you're using Google's generative media models, you don't need to build complex watermarking systems from scratch. Google DeepMind's SynthID, which adds an imperceptible mathematical signature directly into the asset, is natively embedded. Google also supports the C2PA standard for cryptographic metadata. Architect your applications to preserve this metadata and surface these content credentials in your user interfaces.

Shared fate indemnification

You need legal certainty before adopting AI. Google Cloud offers a unique two-pronged generative AI indemnity that covers both the training data used to build the models and the generated output created by your users (provided they follow responsible AI practices and not intentionally infringe on IP). With this legal backing, you can give end-users the compliance and security assurances needed to operate at an enterprise level.





All Google Cloud models capture the cryptographic provenance of a generated asset.

Digital IP and version control

Generative media fundamentally changes how creative assets are stored and recalled. Architectures must be built to version-control the recipe, not just the render. A true "media-aware" knowledge base stores the final MP4 alongside the exact prompt template, the specific API model version, the reference image hashes, and the `GEMINI.md` context files. This way, a brand's visual DNA or a specific character's likeness can be perfectly recalled, modified, and regenerated months later without drifting from the original aesthetic.

Media generation and responsible AI

To ensure a safe and responsible experience, Google's media generation capabilities are equipped with a multi-layered safety approach. This is designed to prevent the creation of inappropriate content, including sexually explicit, dangerous, violent, hateful, or toxic material.

Learn more about verifying content

Technique library

Startups can reference pre-built agentic blueprints to rapidly deploy production-grade generative media pipelines. These examples illustrate how to wire together Google’s models, orchestration tools, and prompting frameworks.

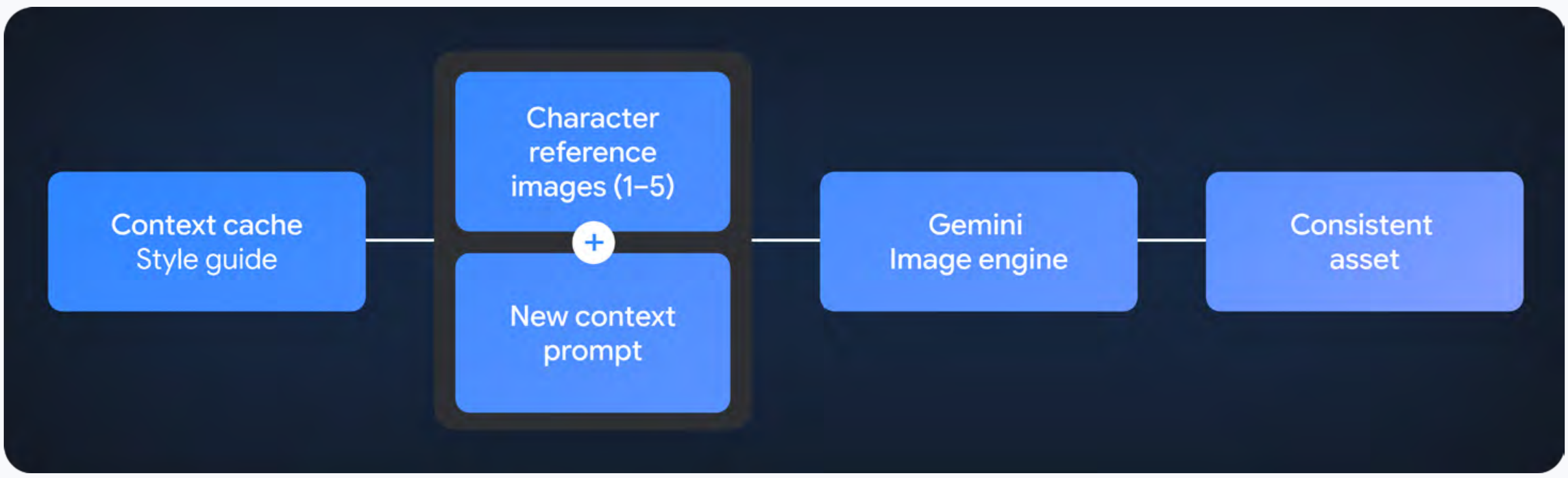
Build a brand arcade

Goal

Generate hundreds of localized marketing images featuring the exact same brand mascot or human talent, wearing different outfits in diverse global locations, without model retraining or fine-tuning.

Core stack

Gemini (reasoning), Nano Banana Pro (rendering).



Architecture

1. **Context cache**
   A developer creates a `GEMINI.md` file containing the brand's style constraints and loads it into Gemini's context window.

2. **Multimodal asset loading**
   The application loads three to five reference images of the character/mascot into memory.

3. **Generation loop**
   For each target location, the application constructs a natural language prompt using the strict [Subject] + [Context] + [Style] template.

   Example prompt: A hyper-realistic photograph of [Reference Images 1–3], wearing a yellow raincoat, standing on a slick cobblestone street in London during a rainstorm, shot on 35mm lens with cinematic lighting.

4. **Native fusion**
   The `google-genai` SDK sends the text prompt and the reference images simultaneously to Nano Banana Pro. The model natively fuses the physical identity of the reference images with the newly prompted environment and clothing, returning a brand-consistent asset.

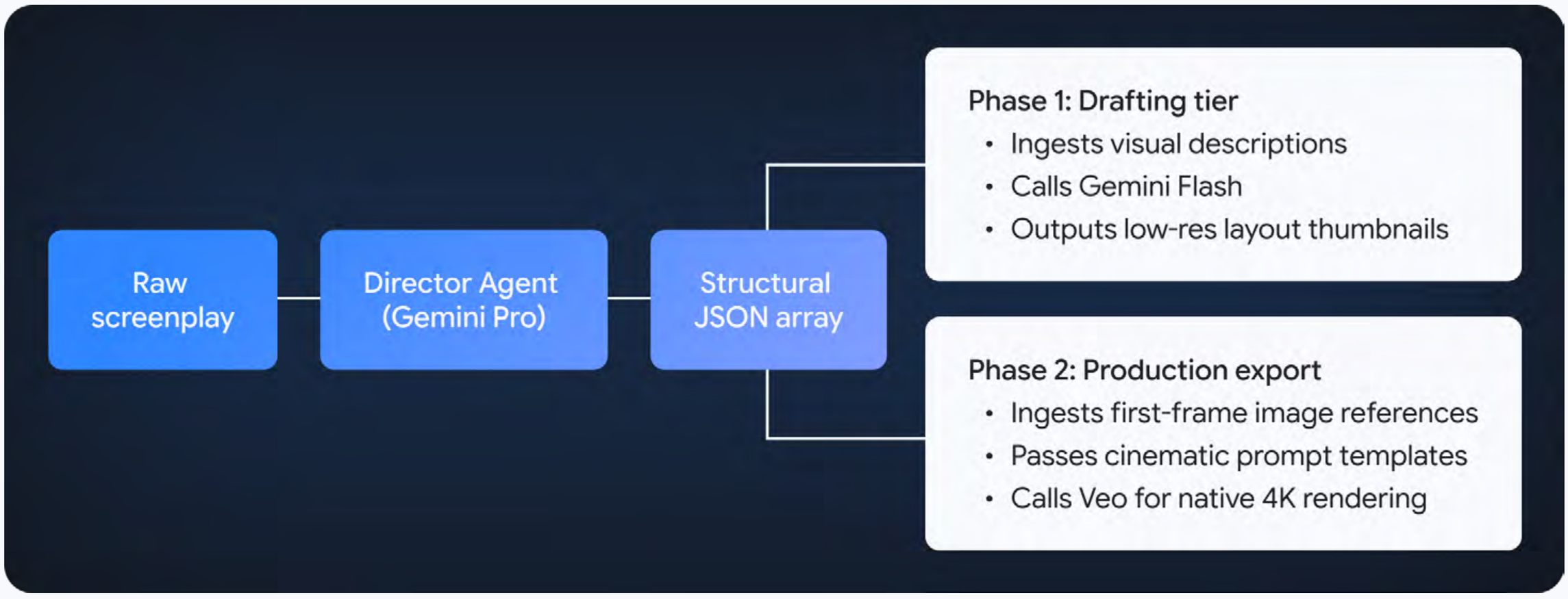
Build an autonomous storyboarding tool

Goal

Convert a raw text script into a fully rendered, timed video sequence with synchronized dialogue and sound effects, operating entirely autonomously.

Core stack

Agent Development Kit (ADK), Gemini (Director Agent), Veo (video and audio render).



Architecture

1. **Script parsing (the Director Agent)**
   The user uploads a script. The Director Agent (powered by Gemini) analyzes the narrative and breaks it down into a structured array of individual shots.

2. **Prompt compilation**
   For each shot, the agent automatically compiles a Veo prompt using the prescriptive [Cinematography] + [Subject] + [Action] + [Context] + [Style & Ambiance] template. It also extracts dialogue and sound effect cues from the script and maps them to temporal timestamps.

3. **Preview/Render split**
   Phase 1: The agent calls an image model (like Nano Banana) to generate a single, low-latency storyboard frame for each shot. These are presented to the user in the UI for approval.
   Phase 2: Upon user approval, the agent forwards the approved image (as a first-frame reference), the parsed template prompt, and the audio cues to Veo.

4. **Final assembly**
   Veo generates the cinematic clips with natively synchronized dialogue and ambient audio. The application then stitches these clips together chronologically based on the agent's original shot array.

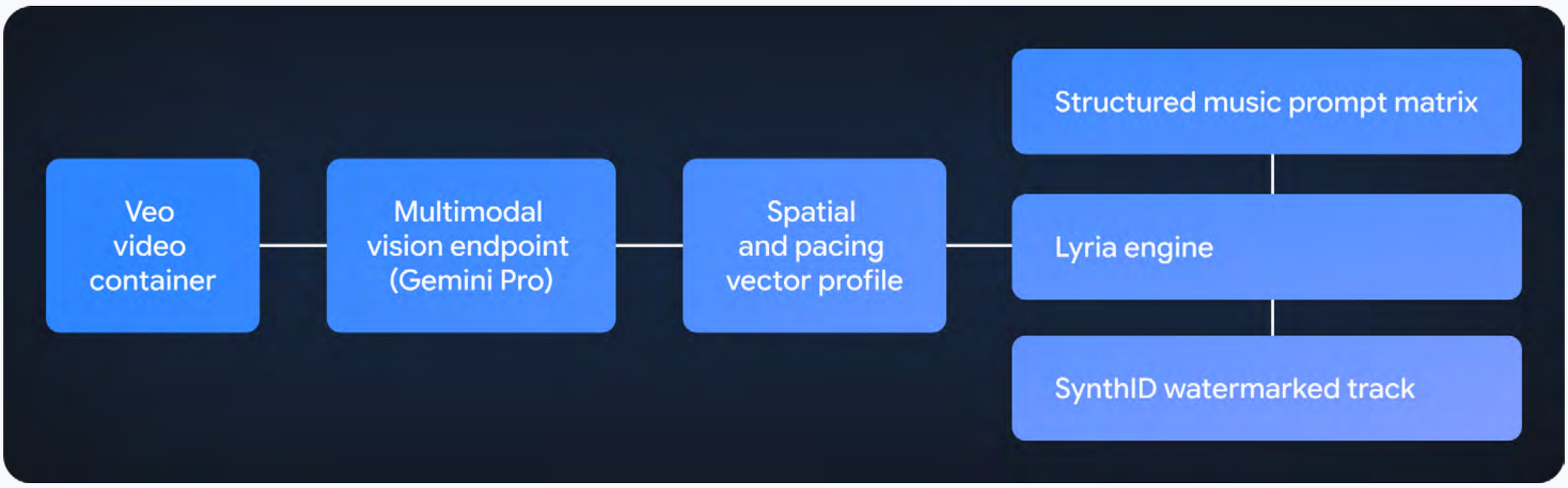
Build a reactive soundtracking tool

Goal

Automatically generate bespoke, royalty-free background music that perfectly matches the emotional arc and pacing of a newly generated video clip.

Core stack

Gemini (vision analysis), Lyria (music generation).



Architecture

1. **Visual emotion parsing**
   Once a Veo video clip is rendered, it is passed back to Gemini's multimodal vision endpoint. The model is prompted to "analyze the visual pacing, color palette, and emotional tone of this video."

2. **Prompt translation**
   The reasoning model translates the visual data into a highly structured music prompt.

   Example prompt: 120 BPM, tense cinematic synth, building to a rapid crescendo, cyberpunk aesthetic.

3. **Audio generation**
   The `google-genai` SDK routes this prompt to Lyria, which produces a high-fidelity, 30-second or 3-minute music track matching the exact granular tempo and emotional mood requested.

4. **Watermarking and storage**
   The final track, natively embedded with a SynthID watermark for AI identification, is muxed with the video file and stored in the asset library.

The horizon is yours

We are just scratching the surface of what language-first orchestration and generative media can achieve. The real breakthroughs won't come from the models themselves, but from how you wire them together to solve real-world creative and technical challenges.

What startups could build next

To get your gears turning, here are some of the most exciting, high-impact agentic architectures we've been hearing about from founders and engineering teams in the ecosystem:

Identity continuity ("One-Shot Arcade")

Maintaining absolute, pitch-perfect character likeness across infinite virtual spaces using Antigravity and Gemini Image (Nano Banana).

Real-time performance tuning

Pushing boundaries with biometrically driven, WebSocket-based music steering—using live heart rate or motion data to guide Lyria via the Python SDK.

Mass-scale batch ad creation

Eliminating creative bottlenecks by parallelizing 4K image generation with real-time web search grounding via the Gemini CLI and Google Search.

Now, we want to hear from you

What are you spinning up? Whether you're trying to optimize an asset-generation loop, build an autonomous agent workflow with ADK, or construct an entirely new interactive experience, you don't have to build in a vacuum.

Get in touch

Google for Startups

Ready to build with Google Cloud?

Get the latest updates in our startup newsletter.

Subscribe

Contact our startup sales team.

Contact us

Learn from Startup School: Generative media.

Watch on-demand

Explore the Future of AI: Perspectives on generative media for startups report.

Read the report

Prefer audio?

Listen to the podcast version of this guide.

Listen now

Made with NotebookLM