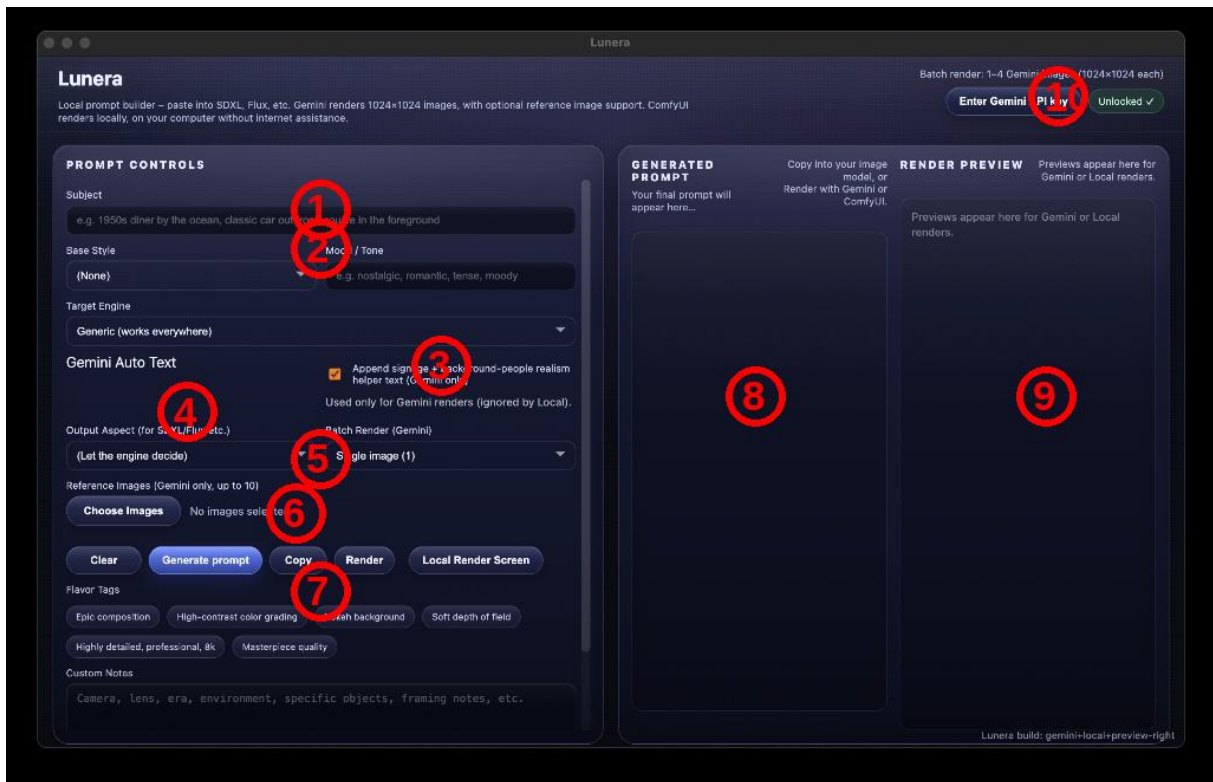


# Lunera User Guide 2026

## 1. Main Screen Controls

The main screen is where prompt building, engine selection, and final render review all come together. The screenshot below shows the primary layout used for cloud rendering and prompt generation.



Main screen screenshot with numbered callouts for the prompt controls, generated prompt area, and render preview panel.

No.	Control	Explanation
1	<b>Subject</b>	The main scene description. A strong Subject field gives every other control better material to work with.
2	<b>Base Style / Mood / Tone / Target Engine</b>	Defines the visual direction, emotional tone, and the engine Lunera should phrase the prompt for.
3	<b>Gemini / cloud options</b>	Controls for realism helper text, aspect guidance, and cloud batch behavior. In updated builds, cloud behavior can now target Gemini or OpenAI depending on engine choice.

No.	Control	Explanation
4	Reference Images	Add image references to steer composition, subject identity, or scene context for compatible cloud workflows.
5	Action Buttons	Includes Clear, Generate Prompt, Copy, Render, and Local Render Screen.
6	Flavor Tags	Fast descriptive tags that enrich the prompt without making you type every phrase manually.
7	Custom Notes	Freeform instructions for camera direction, era, framing, environment, props, and special scene details.
8	Generated Prompt	The final assembled prompt text. If this looks weak, vague, or generic, the render usually will too.
9	Render Preview	This is where cloud or local render results appear after a job runs.
10	API / Unlock Area	Top-right access point for cloud API key handling and unlock status.

## 2. OpenAI Cloud Rendering

OpenAI is now part of Lunera's cloud rendering path. That matters because it gives users a second high-end engine option instead of relying on a single provider. Gemini is still useful for fast iteration and reference-driven experimentation, while OpenAI can be used when you want a more polished finish, stronger lighting coherence, and cleaner commercial-style output.

### How OpenAI fits in Lunera

OpenAI should be described as a cloud rendering engine alongside Gemini, not as a local or separate export path. Users choose the target engine, generate the prompt, and render into the same Render Preview area. The difference is the cloud model being used behind the scenes.

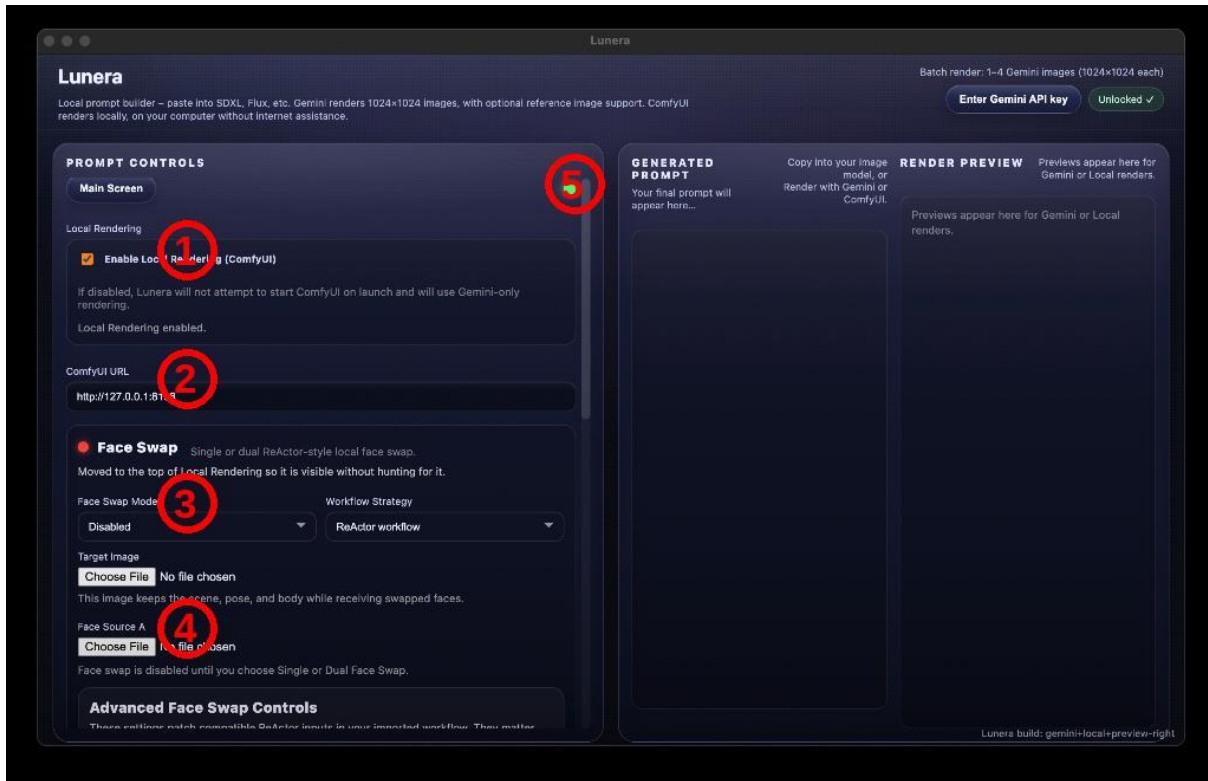
Engine	Best used for	Notes
Gemini	Fast iterations, reference-heavy workflows, roughing out concepts	Strong when you want speed and flexible experimentation.
OpenAI	Higher realism, cleaner commercial imagery, polished final outputs	Best framed as the premium cloud rendering option for refined image quality.

### OpenAI workflow notes

- OpenAI belongs in the same cloud-rendering conversation as Gemini.
- Users still review the final text inside Generated Prompt before clicking Render.
- Renders still appear in the Render Preview panel, so the visual workflow stays familiar.
- OpenAI should be described as a strong option for commercial, photorealistic, and presentation-ready results.

### 3. Local Rendering and Face Swap

The local screen is where ComfyUI integration, face swap controls, and advanced node-mapping settings become visible. This mode is separate from cloud rendering and should remain documented as the path for local workflows, SDXL/Flux pipelines, and face replacement operations.



Local rendering screen showing the ComfyUI toggle, local URL, face swap controls, and the status area.

No.	Control	Explanation
1	Enable Local Rendering	Turns ComfyUI behavior on. If this is off, local render actions will not behave as expected.
2	ComfyUI URL	The local server address Lunera uses to contact ComfyUI. The common default is http://127.0.0.1:8188.
3	Face Swap Mode	Choose the local replacement workflow type, such as single or dual face swap.
4	Advanced Controls	Tuning area for swap models, face restore model, weights, node hints, and matching behavior.
5	Status Light / activity state	Quick visual indicator that the local path is available or active.

#### Practical guidance

- If local rendering appears broken, first confirm the Local Rendering toggle is on.
- Then verify the ComfyUI URL matches the running server.
- Only after that should you start changing workflow JSON, node hints, or swap settings.

## 4. Quick Start and Save Behavior

Lunera is easiest to use when you keep the workflow simple: define the image clearly, choose the right engine, review the prompt, then render. Do not change ten controls at once. Change one thing, test again, and keep moving.

### Quick Start

1. Enter a concrete Subject.
2. Choose Base Style, Mood / Tone, and the correct Target Engine.
3. For cloud work, review the Generated Prompt and click Render.
4. For local work, switch to the Local Render Screen, enable local mode, confirm the ComfyUI URL, and test the path.
5. For OpenAI cloud rendering, keep the same basic flow but describe OpenAI as the refined cloud option for high-end output.

Saving images remains straightforward. Users can drag images directly from Render Preview to the desktop or another folder. Local ComfyUI workflows may also write files into the usual local output folder depending on the workflow configuration.

End of updated guide.