Mobile Artificial Intelligence Distribution Manual

v3.0.0

March 9, 2026

Contents

1 Introduction

Maid (Mobile Artificial Intelligence Distribution) is a free and open-source application for interfacing with AI language models on Android. It supports both fully local inference via `llama.cpp` and remote inference through the APIs of Anthropic, DeepSeek, Mistral, Novita, Ollama, and OpenAI.

All settings are stored locally on your device using encrypted application storage — no API keys or personal data are sent anywhere other than the provider you configure.

1.1 Companion App

For text-to-speech functionality, Maid has a companion app called **Maise**. It can be found at `https://github.com/Mobile-Artificial-Intelligence/maise`.

2 Large Language Model Inference Providers

2.1 Feature Comparison

Provider	API Key	Custom Base URL	Custom Headers	Local	Vision
Llama	No	No	No	Yes	Yes (requires projector)
Ollama	No	Yes	Yes	Yes	Yes (model-dependent)
OpenAI	Yes	Yes	Yes	No	No
Anthropic	Yes	Yes	Yes	No	No
Mistral	Yes	Yes	No	No	No
DeepSeek	Yes	No	Yes	No	No
Novita	Yes	No	Yes	No	No

2.2 Selecting a Provider

On the Settings screen, tap the **API** dropdown at the top of the page to choose which AI provider Maid should use. The available options are:

- **Llama** — Local inference on-device (no internet required)
- **Ollama** — Local or self-hosted server
- **OpenAI** — OpenAI cloud API (and compatible endpoints)
- **Anthropic** — Anthropic Claude cloud API
- **Mistral** — Mistral AI cloud API
- **DeepSeek** — DeepSeek cloud API
- **Novita** — Novita AI cloud API

After selecting a provider, the settings panel will update to show the fields relevant to that provider.

3 Llama (Local Inference)

3.1 Overview

The Llama provider runs AI models directly on your device using `llama.cpp` via the `llama.rn` library. No internet connection is required and no data leaves your device. Performance depends on your device's CPU and available RAM.

3.2 Selecting a Built-in Model

Maid ships with a curated catalogue of GGUF models that can be downloaded directly from within the app. Open Settings, ensure **Llama** is selected as the API, then tap the **Model** dropdown to browse available models. Selecting a model and quantisation will begin the download automatically.

The built-in catalogue includes the following models:

Model	Parameters	Quantisations Available
Qwen 3.5 0.8B	0.8B	UD-IQ2_M through BF16
Qwen 3.5 2B	2.0B	UD-IQ2_M through BF16
Qwen 3.5 4B	4.0B	UD-IQ2_M through BF16
LFM 2.5 1.2B Thinking	1.2B	Q4_0, Q4_K_M, Q5_K_M, Q6_K, Q8_0, BF16, F16
LFM 2.5 1.2B Instruct	1.2B	Q4_0, Q4_K_M, Q5_K_M, Q6_K, Q8_0, BF16, F16
LFM 2.5 VL 1.6B	1.6B	Q8_0, BF16, F16
Qwen3 4B	4.0B	Q4_K_M, Q5_0, Q5_K_M, Q6_K, Q8_0
Phi 3 Mini 4K Instruct	3.8B	Q4, FP16
TinyLlama 1.1B Chat	1.1B	Q2_K through Q8_0
Gemma 2 2B IT	2.0B	IQ3_M through F32
Gemma 3 1B IT	1.0B	Q2_K through BF16
Gemmasutra Mini 2B v1	2.0B	Q2_K through F32
Gemmasutra Small 4B v1a	4.0B	Q2_K through Q8_0
Qwen2.5 1.5B Instruct	1.5B	Q2_K through FP16
Llama 3.2 1B Instruct	1.0B	IQ3_M through F16
Llama 3.2 3B Instruct	3.0B	IQ3_M through F16
Tesslate Tessa T1 3B	3.0B	IQ2_M through BF16

3.2.1 Choosing a Quantisation

Quantisation reduces model size at the cost of some accuracy. A general guide:

- **Q2_K / Q3_K** — Smallest size, lowest quality. Useful when RAM is very limited.
- **Q4_K_M** — Good balance of size and quality. Recommended for most devices.
- **Q5_K_M / Q6_K** — Higher quality, larger download. Suitable for higher-end devices.
- **Q8_0** — Near-lossless quality. Requires the most RAM.
- **F16 / BF16** — Full precision. Not recommended for mobile inference.

3.3 Loading a Custom Model File

To use a GGUF model file you have obtained separately:

1. Transfer the `.gguf` file to your device (e.g. via USB or a file manager).
2. In Maid, go to **Settings** and select **Llama** as the API.
3. Tap **Add Model File** and navigate to the file location using the file picker.
4. Once added, select the file from the **Model** dropdown.

3.4 Vision Support (Multimodal)

The Llama provider supports image input for vision-capable models. To enable this, you must load a **multimodal projector** file alongside the base model.

3.4.1 What is a Projector?

A multimodal projector (also called a clip or vision adapter) is a small auxiliary model file that bridges the image encoder and the language model. It is distributed separately from the base GGUF model and typically has the file extension `.mmproj` or `.gguf`.

3.4.2 Loading a Projector File

1. Transfer the projector file to your device.
2. In Maid, go to **Settings** and select **Llama** as the API.
3. Tap **Add Projector File** and select the file using the file picker. Both `.mmproj` and `.gguf` files are accepted.
4. The projector is automatically linked to the currently active model.

The projector must correspond to the loaded base model — mismatched pairs will not produce useful results. Once a compatible projector is loaded and vision is confirmed, the image attachment button becomes active in the prompt input bar.

3.4.3 Projector Matching Rules

The projector is used when:

- The selected projector key matches the selected model key exactly, **or**
- The model was loaded from a local file (model key ends in `(local)`).

3.5 Configuration

No API key or server address is needed. The only configurable items are:

- **Model** — The GGUF model file to load.
- **Projector** — Optional multimodal projector file for vision support (see Section 3.4).
- **Parameters** — Optional inference parameters (see Section 12.1).

4 Ollama

4.1 Overview

Ollama is a tool for running large language models locally on a desktop or server. Maid connects to an Ollama instance over the network, which means you can run a capable model on a more powerful machine (such as a desktop PC or a home server) and use Maid on your phone as the interface.

4.2 Installing Ollama

Install Ollama on the host machine by following the official instructions at `https://ollama.com`. Once installed, start the Ollama service and pull a model, for example:

```
ollama pull llama3.2
```

By default Ollama listens on `http://localhost:11434`.

4.3 Allowing Remote Connections

If Maid is running on a different device than Ollama, you must allow Ollama to accept connections from other hosts. Set the environment variable before starting Ollama:

```
OLLAMA_HOST=0.0.0.0 ollama serve
```

Ensure your firewall allows inbound connections on port `11434` from your phone's network.

4.4 Find Ollama

Instead of entering the Base URL manually, tap the **Find Ollama** button to have Maid scan your local network automatically. Maid determines the device's current IP address, derives the subnet, and simultaneously probes every host on that subnet at port `11434`. The first responding host is set as the Base URL. An alert is shown if no Ollama instance is found.

This requires both the phone and the Ollama host to be on the same local network, and Ollama must be configured to accept remote connections (see Section above).

4.5 Configuration in Maid

1. Go to **Settings** and select **Ollama** from the API dropdown.
2. Enter the **Base URL** of your Ollama server manually, or tap **Find Ollama** to detect it automatically (see above). Example URL:

   ```
   http://192.168.1.100:11434
   ```

3. Maid will automatically fetch the list of models available on your Ollama server.
4. Select a model from the **Model** dropdown.
5. Optionally configure **Custom Headers** and **Parameters** (see Sections 12.2 and 12.1).

4.6 Vision Support

Vision support in Ollama is **automatic** — Maid queries the selected model's capabilities when you choose a model. If the model reports vision support (e.g. `llava`, `bakllava`, `moondream`, `llama3.2-vision`), the image attachment button is enabled in the prompt input bar. No additional configuration is required.

To use a vision-capable model, pull it with Ollama on the host machine and then select it from the **Model** dropdown in Maid:

```
ollama pull llava
```

4.7 Required Fields

Field	Required	Default
Base URL	Yes	—
Model	Yes	—

5 OpenAI

5.1 Overview

The OpenAI provider connects to OpenAI's cloud API. It also supports any third-party API that is compatible with the OpenAI REST specification, such as LM Studio, vLLM, or OpenRouter, by changing the Base URL.

5.2 Obtaining an API Key

1. Create an account at `https://platform.openai.com`.
2. Navigate to **API Keys** in your account dashboard.
3. Click **Create new secret key** and copy the generated key.

Keep your API key secure. Charges are incurred based on token usage.

5.3 Configuration in Maid

1. Go to **Settings** and select **OpenAI** from the API dropdown.
2. Paste your API key into the **API Key** field.
3. Maid will automatically fetch and populate the **Model** dropdown with the models available on your account.
4. Select a model (e.g. `gpt-4o`, `gpt-4o-mini`, `o3-mini`).
5. Optionally set a **Custom Base URL** if you are using a compatible third-party endpoint.
6. Optionally configure **Custom Headers** and **Parameters** (see Sections 12.2 and 12.1).

5.4 Required Fields

Field	Required	Default
API Key	Yes	—
Model	Yes	—
Base URL	No	`https://api.openai.com/v1`

5.5 Using Compatible APIs

Because the Base URL is configurable, OpenAI mode also works with any OpenAI-compatible endpoint. Examples:

- **LM Studio** — set Base URL to `http://<host>:1234/v1`
- **OpenRouter** — set Base URL to `https://openrouter.ai/api/v1` and use your OpenRouter API key
- **vLLM** — set Base URL to `http://<host>:8000/v1`

6 Anthropic

6.1 Overview

The Anthropic provider connects to Anthropic's Claude family of models via the official Claude API. Claude models are known for their strong reasoning, instruction-following, and safety characteristics.

6.2 Obtaining an API Key

1. Create an account at `https://console.anthropic.com`.
2. Navigate to **API Keys** in the left-hand sidebar.
3. Click **Create Key**, name it, and copy the generated key.

API usage is billed per token. Review Anthropic's pricing page for current rates.

6.3 Configuration in Maid

1. Go to **Settings** and select **Anthropic** from the API dropdown.
2. Paste your API key into the **API Key** field.
3. Maid will automatically fetch and populate the **Model** dropdown with available Claude models (e.g. `claude-opus-4-6`, `claude-sonnet-4-6`, `claude-haiku-4-5`).
4. Select a model.
5. Optionally set a **Custom Base URL** if routing through a proxy.
6. Optionally configure **Custom Headers** and **Parameters** (see Sections 12.2 and 12.1).

6.4 Required Fields

Field	Required	Default
API Key	Yes	—
Model	Yes	—
Base URL	No	`https://api.anthropic.com`

7 Mistral

7.1 Overview

The Mistral provider connects to Mistral AI's cloud API. Mistral offers a range of efficient and capable models including Mistral Large, Mistral Small, and Codestral.

7.2 Obtaining an API Key

1. Create an account at `https://console.mistral.ai`.
2. Navigate to **API Keys** in the left-hand menu.
3. Click **Create new key** and copy the generated key.

7.3 Configuration in Maid

1. Go to **Settings** and select **Mistral** from the API dropdown.
2. Paste your API key into the **API Key** field.
3. Maid will automatically fetch and populate the **Model** dropdown with available Mistral models.
4. Select a model (e.g. `mistral-large-latest`, `mistral-small-latest`).
5. Optionally set a **Custom Base URL** if using a self-hosted Mistral-compatible server.
6. Optionally configure **Parameters** (see Section 12.1).

Note: The Mistral provider does not support custom HTTP headers.

7.4 Required Fields

Field	Required	Default
API Key	Yes	—
Model	Yes	—
Base URL	No	`https://api.mistral.ai`

8 DeepSeek

8.1 Overview

The DeepSeek provider connects to the DeepSeek cloud API. DeepSeek offers high-quality models at competitive pricing, including the DeepSeek-V3 and DeepSeek-R1 reasoning model.

8.2 Obtaining an API Key

1. Create an account at `https://platform.deepseek.com`.
2. Navigate to **API Keys** in your dashboard.
3. Click **Create API Key** and copy the generated key.

8.3 Configuration in Maid

1. Go to **Settings** and select **DeepSeek** from the API dropdown.
2. Paste your API key into the **API Key** field.
3. Maid will automatically fetch and populate the **Model** dropdown.
4. Select a model (e.g. `deepseek-chat`, `deepseek-reasoner`).
5. Optionally configure **Custom Headers** and **Parameters** (see Sections 12.2 and 12.1).

Note: The DeepSeek provider always connects to `https://api.deepseek.com`. The base URL is not configurable.

8.4 Required Fields

Field	Required	Default
API Key	Yes	—
Model	Yes	—

9 Novita

9.1 Overview

The Novita provider connects to Novita AI's cloud API. Novita provides high-performance inference for a wide variety of open-source models through an OpenAI-compatible interface.

9.2 Obtaining an API Key

1. Create an account at `https://novita.ai`.
2. Navigate to your dashboard and generate an API key.

9.3 Configuration in Maid

1. Go to **Settings** and select **Novita** from the API dropdown.
2. Paste your API key into the **API Key** field.
3. Maid will automatically fetch and populate the **Model** dropdown.
4. Select a model from the available list.
5. Optionally configure **Custom Headers** and **Parameters** (see Sections 12.2 and 12.1).

Note: The Novita provider always connects to `https://api.novita.ai/openai`. The base URL is not configurable.

9.4 Required Fields

Field	Required	Default
API Key	Yes	—
Model	Yes	—

10 Prompt Input

The prompt input bar is located at the bottom of the chat screen. It consists of a multiline text field and a context-sensitive action button on the right.

10.1 Text Field

Tap the text field to open the keyboard and type a message. The field grows vertically as you type to accommodate longer messages. While the field is empty its placeholder reads *"Type a message..."*.

10.2 Clear Prompt

When the text field contains any text, a **Clear Prompt** link appears above the input bar. Tapping it instantly empties the field.

10.3 Action Button

The button on the right of the input bar changes depending on the current state:

- **Send** — Shown when the field contains text. Tap to submit the message and start a model response. The button is disabled if the selected provider is not ready (e.g. no model loaded or no API key set).
- **Stop** — Shown while the model is generating a response. Tap to cancel the in-progress generation.
- **Microphone** — Shown when the field is empty and the model is not generating. Tap to begin voice dictation (see Section 10.6).
- **Microphone off** — Shown while voice dictation is active. Tap to stop listening; the transcribed text is appended to the field.

10.4 Attaching Images

When the active provider and model support vision (currently **Llama** with a projector loaded and **Ollama** with a vision-capable model), an **image** icon appears to the left of the action button in the prompt input bar. The button is disabled when vision is not available.

To attach images:

1. Tap the **image** icon. Maid will request photo library access on first use.
2. Select one or more images from your photo library.
3. The selected images appear as thumbnails in a horizontal strip above the input field. Tap the × on any thumbnail to remove it before sending.
4. Type your message (optional) and tap **Send**. The images are included with your message and passed to the model alongside the text.

Images are encoded as base64 and attached to the message metadata. They are sent only to the configured provider and are not stored or transmitted anywhere else.

10.5 Sending a Message

Type your message and tap **Send** (or tap the microphone to dictate it). Maid appends a user message node and an empty assistant message node to the conversation tree, then streams the model response into the assistant node in real time. The input field is cleared immediately after the message is submitted.

If this is the first message in a new chat, a system prompt node is created automatically using the system prompt configured in Settings (defaulting to *"You are a helpful assistant."*).

10.6 Voice Dictation

Tap the **microphone** icon to dictate your message by voice. Maid will request microphone permission on first use; if permission has been permanently denied, dictation will not start. Once listening begins, the button switches to a **microphone off** icon which you can tap to stop early. When recognition finishes, the transcript is appended to any text already in the field, with a space inserted between existing text and the new transcript.

Dictation uses the device's on-board speech recognition engine and is performed in English (en-US) with punctuation added automatically.

11 Messages

Each message in the chat view has a row of controls displayed in the top-right corner of the message. The controls available depend on whether the message is from the user or the assistant.

11.1 Branch Navigation

Every message in Maid is part of a tree rather than a flat list. When a response is regenerated or a user message is edited, a new branch is created rather than overwriting the existing message. The branch navigator lets you move between these branches.

- **Left arrow** (`menu-left`) — Switch to the previous sibling of this message.
- **Counter** — Displays the current branch position as *current* / *total* (e.g. `2 / 3`).
- **Right arrow** (`menu-right`) — Switch to the next sibling of this message.

The arrows are disabled when there is only one branch, when a message is being edited, or while the model is generating a response.

11.2 Delete

The **delete** (trash) icon is available on all messages. Tapping it removes the message and all of its descendants from the conversation tree. This action cannot be undone, so use it with care.

Delete is disabled while a message is being edited or while the model is busy.

11.3 Assistant Message Controls

The following controls are shown on **assistant** messages only.

11.3.1 Regenerate

Tap the **reload** icon to regenerate an assistant message. A new branch is created under the parent user message and the model is prompted again with the current conversation up to that point, using the active provider and parameters. The existing response is preserved and accessible via the branch navigator.

Regeneration is disabled while another message is being edited or while the model is busy.

11.3.2 Text-to-Speech

If a voice has been selected in Settings, a speaker icon appears on assistant messages. The active voice can be changed at any time from the **Voice** option on the Settings screen. Available voices include those provided by system TTS engines such as Google Text-to-Speech (pre-installed on most Android devices) as well as voices from the Maise companion app.

- **Volume high** icon — Tap to read the assistant message aloud using the selected voice. If the message contains a reasoning block, only the final response (not the internal reasoning) is spoken.
- **Volume off** icon — Shown while speech is playing. Tap to stop playback immediately.

TTS is disabled while a message is being edited or while the model is generating a response.

11.3.3 Feedback (Upvote / Downvote)

Once an assistant message has finished generating, a pair of thumbs icons appears at the bottom of the message content.

- **Thumbs up** — Marks the response as good. The message content, provider name, and model name are submitted to the developer as a positive report.
- **Thumbs down** — Marks the response as poor. The same information is submitted as a negative report.

Reports are submitted anonymously via Supabase. If no existing session is found, an anonymous sign-in is performed automatically before the report is inserted. A confirmation dialog is shown on success, or an error alert if the submission fails.

No personally identifying information beyond the anonymous session identifier is attached to the report. The purpose of these reports is to help the developer assess model quality across different providers and configurations.

11.4 User Message Controls

The following control is shown on **user** messages only.

11.4.1 Edit

Tap the **pencil** icon to edit a user message. The message content becomes editable in-place. Submitting the edited message creates a new branch from the parent of the original message, preserving the original text in the previous branch. The branch navigator can then be used to switch between the original and the edited version.

Editing is disabled while another message is already being edited or while the model is generating.

12 Advanced Configuration

12.1 Model Parameters

All providers expose a **Parameters** panel that lets you fine-tune inference behaviour. Tap **Parameters** on the Settings screen to expand the panel. Common parameters include:

Parameter	Description
`temperature`	Controls randomness. Lower values (e.g. 0.2) produce more deterministic output; higher values (e.g. 1.0) produce more varied output.
`top_p`	Nucleus sampling threshold. Limits token selection to the top p cumulative probability mass.
`top_k`	Limits token selection to the top k most likely tokens at each step.
`max_tokens`	Maximum number of tokens the model may generate in a single response.
`seed`	Sets a fixed random seed for reproducible outputs (where supported).

Parameters that are left blank are omitted from the API request, allowing each provider to use its own defaults.

12.2 Custom HTTP Headers

The OpenAI, Anthropic, DeepSeek, Novita, and Ollama providers allow you to supply additional HTTP headers. This is useful for:

- Passing organisation or project identifiers required by a corporate proxy.
- Supplying additional authentication tokens (e.g. `HTTP-Referer` required by OpenRouter).
- Tagging requests for logging or rate-limit tier purposes.

To add a header, tap **Headers** on the Settings screen, then tap the + button. Enter the header name and value, then save. Multiple headers can be added.

13 Signing Fingerprints

To verify that an APK was signed by the official Maid developer, check its certificate fingerprints against the values listed here. A mismatch indicates the APK may have been tampered with or repackaged.

13.1 Signing Key

The release signing key fingerprints are:

Algorithm	Fingerprint
MD5	`BE:AC:29:41:F5:41:D2:26:42:DD:D1:A3:85:21:E1:16`
SHA-1	`48:F6:DC:73:09:CE:19:C6:A9:70:7E:A2:9A:B7:6F:42:2D:41:32:30`
SHA-256	`83:5E:D2:2E:D8:95:C4:C2:72:D6:98:AA:6E:4E:48:DB:` `0B:4E:36:DC:CF:70:10:D5:DE:15:03:4A:C9:E1:B9:6F`

13.2 Upload Key (Google Play)

APKs distributed through Google Play are re-signed by Google using the upload key. Its fingerprints are:

Algorithm	Fingerprint
MD5	`C0:86:A0:F3:E8:E5:4D:46:60:8A:37:4E:DB:11:CC:C7`
SHA-1	`77:FC:77:2B:21:5E:9F:36:31:79:09:DF:7D:F4:1F:CA:96:0C:39:17`
SHA-256	`54:EE:B9:9F:14:38:D9:68:9B:C2:C6:7F:F9:DD:A3:E3:` `D8:28:D3:80:76:46:B7:24:46:71:9F:61:D9:63:E6:98`

13.3 How to Verify

To inspect the certificate of a locally built or sideloaded APK, run the following command using the Android SDK build tools:

```
apksigner verify --print-certs maid.apk
```

Compare the printed SHA-256 fingerprint against the signing key value above. For APKs downloaded from Google Play, compare against the upload key fingerprint instead.