

User Guide

Early access is by contacting the project. No public installer or purchases are available. Use the access address and instructions supplied by your operator.

Use Friday to explain each step; compare its guidance with your screen and verify the result. Available controls depend on your role and deployment. This guide is application guidance, not a release certification.

Online help: fridaylocalai.com/help

Contact: fridaylocalai.com/contact

In this guide

1. Help & How-To Guide
2. Workspace & Organizer
3. Choose a Model
4. Memory & Context
5. Agents
6. Skills & Tools
7. Knowledge System
8. Media Studio
9. Diagnose a Failed Response

1. Help & How-To Guide

A little clarity goes a long way.

Find a practical starting point for setup, conversations, files, models, and troubleshooting.

Check your environment

Confirm authorized sign-in, model availability, storage configuration, and a simple saved conversation.

Start a conversation

State a goal, constraints, context, and the form of the answer you need.

Verify a document

Check both accepted upload and actual content access.

Choose a model

Match supported inputs and resource needs to the task.

Diagnose a failure

Preserve evidence and isolate the failing layer before changing services.

Let Friday help you learn FridayLocalAI

You can ask application questions directly in a conversation. Relevant active Help topics may be supplied to the response when available to your role. You can also use the information icons beside Organizer, Projects, Conversations, and other sections. A reply is guidance: compare it with the controls and results in your installation.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

I am new to FridayLocalAI. Guide me through a small practice task one step at a time. Ask what I want to do, use the relevant built-in Help where available, explain each control before I use it, and wait for me to confirm the result.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know your current settings.

Choose the kind of help you need

Start by deciding whether you need an explanation, a guided action, or help with an error. An explanation answers a question such as "What is a conversation scope?" A guided action walks you through something you can do in the interface. Troubleshooting investigates why the expected result did not occur. Naming the kind of help you want makes it easier for Friday to respond at the right level. Describe your goal in ordinary language and mention the section you are using. You do not need to know an internal component name. "I am in Organizer and cannot find yesterday's conversation" is a more useful starting point than "the system is broken." Include what you already tried, but keep private content and account secrets out of the description.

Use the information beside the interface

The application includes information icons associated with workspace sections and controls. These can provide explanations relevant to the area you are viewing. An icon beside a conversation description may explain that item's context, while a section-level Help control can explain the section itself. Read the label and the resulting explanation rather than assuming every icon has the same purpose. If the guidance does not answer your question, bring the specific wording into a conversation with Friday. State which part remains unclear and what you are trying to accomplish. Relevant active Help topics may be supplied to the response when available to your role. A missing or unhelpful explanation is useful feedback; it is not a reason to invent a control or make an administrative change.

Ask for one step at a time

A guided exchange is easiest to follow when Friday gives one action and an expected result, then waits for you. After completing the action, describe what changed. If nothing changed, say that explicitly. If your screen differs from the instructions, provide the visible labels rather than guessing which button might be equivalent. For example, ask how to start a practice conversation. Friday can explain where to look in Organizer and what creation controls may be available. You perform the permitted action and confirm the result. This keeps the distinction between guidance and execution clear. It also prevents a long chain of instructions from carrying you past an early mismatch that should have been resolved first.

Provide enough context without oversharing

Useful context includes the feature, your intended task, the visible error if any, and the point at which you became uncertain. A short description of the Agent and model selection may matter for a response issue. A source's type and approximate size may matter for an attachment question. The contents of unrelated documents, credentials, and private account details usually do not help. Use a small fictional example when possible. If a document question can be reproduced with three harmless sentences, begin there. If the problem only occurs with a particular source, describe that distinction to your operator and agree on a suitable way to review it. The public contact route is not an invitation to send your entire workspace or raw diagnostic material.

Decide whether the guidance worked

Before following a step, understand what success should look like. Opening a panel should reveal the intended controls. Sending a message should produce the expected conversation entry. Saving a note should make the note visible in the correct scope. Completing a source review should produce an answer that can be checked against the source. If Friday reports a successful change but you cannot find the result, ask for clarification and inspect the relevant screen. Do not accept a conversational assurance as a substitute for evidence. For factual answers, compare the important details with the original material. For a workflow you will repeat, record the short sequence that worked so you can return to it without rediscovering every step.

Get help when conversation is unavailable

If sign-in or chat is failing, the application cannot be your only support channel. Use the first-run and troubleshooting pages on this website, or keep the downloadable consumer guide available. Contact your operator with a concise description of the failure, the time, what you expected, and what still works. Avoid repeated retries that could duplicate an external action or overwrite useful evidence. Access questions go through the project's early-access contact path. There is no public installer to download and no store purchase that automatically creates a workspace account. Once access is established, return to a harmless practice task before important work. The purpose of these guides is to help you understand the application well enough to use it deliberately and recognize when further assistance is needed.

Web guide: fridaylocalai.com/help

2. Workspace & Organizer

Give ongoing work a home.

Projects, folders, conversations, and search provide structure for work that continues beyond a single prompt.

Projects and folders

Use a project for a sustained objective and folders to group related work. Access and creation controls depend on the user's role; visibility is not permission to change everything.

Conversations

Begin with the task, constraints, and the result you need. Keep the discussion with the project it belongs to and check saved history when returning to the work.

Search and context

The Organizer includes search across projects, folders, conversations, and messages. Check which project and conversation are active before adding documents or changing contextual guidance.

Dashboard and workflow

The Dashboard includes project context, a workflow-board interface, Skills & Tools, and memory summaries. Use the information available in your deployment to understand the task's boundaries.

Find the right place to work

Use Toggle Organizer if the panel is hidden. The search field covers projects, folders, conversations, and messages. Check the selected project and conversation before sending content. Use Create new conversation when your role permits it; project creation can require an administrator.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Help me organize [goal] in FridayLocalAI. Suggest a simple project, folder, and conversation structure, then guide me through the controls available to my role. Do not claim anything was created until I verify it in Organizer.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know your current settings.

Organize around outcomes

Use a project for a continuing area of work rather than creating a new structure for every prompt. A project might hold an article series, a research question, or an internal process improvement. Folders can separate useful areas within that project, and conversations can focus on individual questions or deliverables. Keep the structure simple enough that you can explain why each part exists. For example, a research project might have conversations for source questions, comparison notes, and an eventual summary. It does not need a folder for every short exchange. Start with the smallest arrangement that makes the work easy to find, then add organization when there is a clear need. A tidy structure is useful only if it supports the way you actually return to and review the work.

Confirm the selected context

Before sending a message, inspect the active conversation title and its associated context. This matters especially after searching, switching projects, or returning to the application. A response belongs to the conversation in which it was requested, and guidance saved in a particular scope may affect related work. Taking a moment to confirm the target prevents avoidable confusion later. Do the same before attaching a file or creating a Memory Note. If the selected item is not what you expected, navigate to the intended work first. Ask Friday to explain the visible structure if necessary, but do not assume it can see every item on your screen. Describe the names and labels relevant to your question without exposing unrelated private project information.

Use meaningful names and descriptions

A useful conversation title describes the task: "Draft the customer onboarding outline" is easier to recognize later than "New chat" or the name of a model. If renaming is available to your role, use it once the purpose becomes clear. Descriptions can explain the intended scope, such as the audience, source set, or decision the work supports. Keep names short enough to scan and descriptions focused enough to remain useful. Avoid embedding credentials or sensitive personal information in labels. A title is navigation metadata, not an access control or a complete instruction. If the assistant needs a detailed requirement, include that requirement in the appropriate conversation or prompt context rather than expecting a name alone to communicate the whole task.

Search for the work you remember

Organizer search includes projects, folders, conversations, and messages. Start with a distinctive term from the task, source, or expected result. Review the result in context before continuing work. Similar wording can occur in several conversations, so confirm the title and the relevant exchange rather than assuming the first result is the one you need. If the panel is hidden, use the workspace layout control to show Organizer. If a search does not find the expected item, check the wording and the intended workspace with the controls available to you. A search result is not permission to access another user's material, and a missing result does not prove deletion. Preserve the distinction between navigation, visibility, and the underlying state of the work.

Keep conversations focused but connected

A conversation works best when its participants can understand the current objective and the context needed to continue. If the task changes substantially, consider starting a separate conversation in the appropriate place. If it remains the same task, give a concise update rather than repeating the entire history. State the decision made, the new information, and the next result you want. Ask Friday to summarize the current state when a discussion becomes long: established facts, open questions, decisions, and next steps. Review that summary before using it as a handoff. A generated recap can omit or misstate details, so it should not silently replace the original exchange. For enduring guidance, consider a Memory Note with an appropriate scope rather than relying on an unreviewed summary.

Work within the controls available to you

Creating a project and creating a conversation can have different role requirements. If a control is absent or unavailable, ask the operator about the intended workflow. Do not treat the limitation as something a prompt should bypass. A shared workspace needs consistent rules for who can organize, modify, or administer it. When you return to a task, verify that the expected conversation and recent messages are present. If state appears inconsistent, stop making organizational changes and record what you observed. Friday can help explain the interface or prepare a report, but the saved structure is the evidence of what exists. Good workspace habits combine clear naming, a manageable hierarchy, intentional context, and a reliable way to recognize the work you meant to open.

Web guide: fridaylocalai.com/workspace

3. Choose a Model

The right model starts with the input.

Text, document analysis, vision, and generation workflows can require different capabilities and services.

Identify the requirement

Determine the input type and desired output. An available model is not necessarily suitable for every task.

Check the resources

Review the model's own requirements and the deployment's capacity. Larger models and longer contexts can increase resource demand.

Verify delivery

For image understanding, confirm both vision support and delivery of the actual image. A filename in a prompt does not establish image access.

Test a representative task

Use a short, checkable example after changing the model. If it fails, inspect model-service status and configuration before treating the result as a quality problem.

Requested model and active model

The Model selector expresses your requested model for future replies. Agent or capability routing can select a different active model. Check the model attribution on the actual reply when comparing results. Changing the selector does not rewrite earlier messages or guarantee that a particular input type is supported.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Explain the requested Model selector and the active model attribution in FridayLocalAI. Help me design one small test for [text/document/image task] and tell me what to check in the reply. Do not claim a model was used without attribution evidence.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know your current settings.

Start with the input and result

Choosing a model begins with the task you want to complete. Are you asking for an explanation, editing a paragraph, analyzing readable document text, interpreting an image, or using a separate generation workflow? These are different requirements. A model that is useful for one kind of input may not support another, and a visible option does not establish that every associated capability is configured. Describe the task before choosing based on a name or an advertised size. Identify the input you will provide and how you will check the answer. If the task depends on a source, confirm that the application can supply the relevant contents. A model-selection decision cannot compensate for a document that never became available to the response process.

Read requested and active attribution separately

The Model selector represents the requested choice for future replies. FridayLocalAI's Agent and capability routing may influence which model actually produces a response. The active attribution shown with the conversation or reply is therefore important evidence. Check it when a result is unexpected or when you are comparing models. Earlier replies retain their own history; changing a selector does not retroactively regenerate them. If you want a comparison, make a new bounded request and record its actual attribution. Keep the prompt, source material, and review criteria comparable. Otherwise, differences in the task or context can be mistaken for differences in the model, leading to a conclusion that the evidence does not support.

Consider resources without assuming a universal rule

A local LLM uses resources available to its deployment. Model size, context, simultaneous activity, and the selected execution path can affect suitability. The application includes model-management and safety-related controls, but the options available to you depend on the installation and your role. Ask the operator about a supported choice when the requirements are unclear. Do not assume the largest available model will always give the most useful answer. A smaller option may be appropriate for a straightforward task, while a more demanding task may require a different capability or better source context. The right comparison includes accuracy, usability, responsiveness, and the amount of correction needed. Hardware specifications alone do not settle the quality of a particular result.

Run a small comparison you can evaluate

Choose a task with an answer or review standard you understand. For example, supply a short paragraph with known facts and ask for a structured summary without added details. Add a case with an intentionally missing value and check whether the response leaves it unknown. If document work matters, include a narrow source question after verifying attachment access. Record the active model, the prompt, and the outcome for each run. Avoid judging solely by length or confidence. Note whether the answer followed constraints, preserved facts, and acknowledged uncertainty. If a result changes, repeat only as needed to understand the difference. A small, documented comparison is more useful than switching randomly among models and remembering only the most impressive response.

Recognize when the problem is elsewhere

A failed answer can result from missing source content, an unavailable service, an unsuitable input type, or a misunderstood request. It is not always a model-quality problem. Before changing the model, ask what evidence points to the failing stage. Check whether a short text prompt works and whether the issue follows a particular attachment or workflow. If the response claims to see an image but only a filename was supplied, the important question is whether the actual image reached a supported path. If a document summary lacks the requested section, check source access and context. Changing the selected model without addressing those conditions may produce a different-looking answer while leaving the underlying problem unresolved.

Use Friday to plan the decision

Ask Friday to explain the relevant controls and help design a comparison using the options you can see. Supply the model labels and your intended task without expecting it to know the machine's resources automatically. Ask it to identify what remains unknown and what should be confirmed with the operator. If you do not have permission to manage models, use the supported choices made available to your role. Do not download or alter system components based on an improvised conversational instruction. FridayLocalAI early access follows an operator-supported deployment path, and the public website does not provide a universal installation recipe. A good model choice is one you can explain in terms of the task, available capability, observed result, and practical limits of the environment.

Web guide: fridaylocalai.com/help-choose-model

4. Memory & Context

Put guidance where it belongs.

FridayLocalAI's Memory Notes interface distinguishes global, organization, project, folder, and conversation scopes.

Choose the appropriate scope

Use global and organization guidance only when appropriate and permitted, project guidance for an ongoing area of work, folder context for related material, and conversation context for the specific discussion.

Write intentionally

A conversation mentioning a fact is not, by itself, proof that a durable memory was written. Confirm saved notes through the management interface and the relevant scope.

Keep sources separate

A memory note, a document in storage, and context supplied to a model have different purposes. Review provenance and retrieval when an answer depends on source material.

Review and maintain

Keep guidance current and appropriate to the audience and task. The model's response does not establish persistence or future access.

Save and maintain useful guidance

Open Dashboard, find Memory, and choose Manage to open Memory Notes. Where permitted, choose Scope and Category, enter a concise note, and use Add. Confirm the saved note in the list. Existing notes can be edited where permitted; changes are versioned. Keep temporary task details in the conversation rather than making every remark durable.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Help me draft a short Memory Note for this preference: [preference]. Recommend the narrowest suitable scope, explain who or what work it should apply to, and guide me through saving it. Do not say it is saved until I verify the note in Memory Notes.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know your current settings.

Steps to follow

- Choose Conversation for guidance limited to one discussion.
- Choose Folder or Project when the guidance applies to the related work in that scope.
- Organization and Global are broader scopes; use them only when your role and the intended audience make that appropriate.
- Select an appropriate Category such as Preference, Workflow, Fact, or Glossary.

- After adding or editing, read the saved note and its scope back in the interface.

Decide what should last

Memory Notes are useful for guidance that should remain available beyond one exchange. Examples include a writing preference, a project convention, or a definition used repeatedly in related work. Temporary details, tentative guesses, and one-off instructions may be better left in the conversation. Saving everything creates more material to maintain and can make it harder to identify what still applies. Before adding a note, ask whether it is accurate, stable enough to reuse, and relevant to the intended scope. Write it so another authorized participant could understand its meaning without the entire original conversation. A short, specific statement is usually more useful than an unstructured transcript. If a fact is uncertain, preserve that uncertainty instead of converting it into an authoritative-sounding instruction.

Choose the narrowest suitable scope

The Memory Notes interface includes Global, Organization, Project, Folder, and Conversation scopes, with availability governed by role and policy. The broadest choice is not automatically the best one. A formatting requirement for a single deliverable should not silently become guidance for every project. A project-specific term should remain associated with the work that gives it meaning. Consider who and what work should be affected. Use a conversation scope for a local requirement, a project or folder scope for related activity, and broader scopes only when the guidance genuinely belongs there and you are permitted to manage it. Ask Friday to help reason about the choice, then confirm the selected scope in the interface before saving.

Separate memory from sources and prompt layers

A Memory Note is not a substitute for an original document. A saved preference can shape an answer, but it does not prove the model read an attachment or can access a source. Similarly, a prompt layer describes how to approach work but does not grant a Tool permission. Keeping these distinctions clear prevents the application's context mechanisms from being mistaken for one universal store of knowledge and authority. For a source-dependent claim, keep a reference to the relevant source and review the passage when needed. For a recurring behavioral preference, a concise note may be appropriate. For instructions that apply only to the present task, use the conversation. Choosing deliberately makes it easier to explain why information influenced an answer and where a correction should be made.

Verify the saved record

After using Add, inspect the note in the list and check its content, category, and scope. If you edited an existing note, confirm that the updated wording is visible. The application supports versioned changes where permitted, so maintaining a note does not require treating deletion and replacement as the default workflow. Do not rely solely on Friday saying "I will remember that." A conversational promise is not a database record. If you asked for help drafting a note, distinguish the draft from the completed save. You can use Friday to improve clarity or suggest a scope, then carry out and verify the permitted interface action. That sequence keeps guidance useful while making persistence observable.

Review guidance as work changes

A note that was useful at the start of a project may become inaccurate after a decision changes. Review the relevant notes when a workflow, audience, or convention changes. Look for contradictions, outdated instructions, and details that belong in a narrower scope. Update the actual note where permitted rather than repeatedly correcting the assistant in unrelated conversations. Keep edits focused. State the current rule and the conditions under which it applies. Avoid embedding sensitive credentials or an entire private source document in a memory field. If the note needs provenance, include a concise reference that the intended audience can understand. The goal is maintainable context, not an ever-growing accumulation of assertions that nobody has checked.

Understand what memory does not promise

Saving a note is different from training the underlying model. It also does not guarantee that every future task, Agent, or conversation receives that note. Applicable context depends on scope, policy, and the runtime's preparation process. If a response seems inconsistent with expected guidance, inspect the selected context and the note before concluding that it was lost. Ask Friday to explain which guidance is relevant where evidence is available, but do not ask it to invent an account of hidden state. If the interface and response disagree, preserve a small example for the operator. Good memory practice combines deliberate scope, concise wording, verified saves, and periodic review. It supports continuity without replacing source evidence, application permissions, or the user's responsibility to evaluate important results.

Web guide: fridaylocalai.com/memory-and-context

5. Agents

A role with a defined purpose.

Friday's Workshop includes an agent-management panel. Treat agent identity, knowledge, models, permissions, and optional integrations as distinct configuration concerns.

Define the role

Describe the agent's purpose, relevant knowledge, and limits. A specialized label does not guarantee expertise, factual accuracy, or permission to perform actions.

Choose capabilities deliberately

A useful agent needs appropriate model and tool access. Skills describe procedures; tools expose governed operations.

Optional presentation and connections

The agent interface includes voice-assignment and correspondence-related configuration. Availability depends on the enabled services and operator setup; these are not blanket offline capabilities.

Validate the intended work

Use representative tasks to check behavior and boundaries before relying on an agent. Specific preconfigured expert offerings should be listed only with release-specific evidence.

Choose an Agent and inspect the reply

Select an Agent suited to your task where the selector is available. Review the active Agent and model attribution shown with the conversation or reply. A requested choice can be affected by routing and capability rules. Test the role with a small, verifiable task before relying on it.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Help me choose among the Agents I can see: [names and descriptions]. My task is [goal]. Explain the tradeoffs, identify any required capabilities, and suggest a small test. Do not assume any Agent has access to my files.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know

your current settings.

Understand the role before selecting it

An Agent is a configured assistant role within the application. A useful role has a clear purpose, such as helping with a kind of analysis or a recurring workflow. Its description can help you decide whether it suits the task, but a name alone does not guarantee subject expertise, factual accuracy, or access to the resources needed to complete the work. Read the available description and compare it with your intended outcome. If the fit is uncertain, ask Friday to explain the choices you can see. Supply the relevant labels and your goal rather than expecting the assistant to know every available configuration automatically. Begin with a small task whose result you can evaluate before making the Agent part of an important or repeated workflow.

Keep Agent identity separate from the model

The Agent's role and the model producing a response are different parts of the system. An Agent may guide how work is approached, while a model supplies the response capability. FridayLocalAI exposes requested and active attribution so users can inspect the response path rather than assuming every choice maps directly to a single model in every situation. When comparing behavior, record the active Agent and model information shown with the reply. Keep the prompt and source material similar enough that the comparison means something. A different result may come from changed context, routing, or input as well as from the role itself. Avoid assigning all differences to the Agent name without checking what actually happened during the request.

Provide a task brief

Even a specialized Agent needs a clear request. State the goal, relevant context, constraints, and expected output. For a review task, explain what should be checked and what evidence is available. For a drafting task, identify the audience and the form of the result. If important information is missing, ask the Agent to question you before proceeding. For example, a request to compare two approaches should specify the criteria that matter. Otherwise, the answer may choose criteria that are plausible but irrelevant to your decision. A useful brief also identifies what should not happen, such as sending a draft or modifying a source file. This makes the intended boundary visible without pretending that the prompt itself grants or removes runtime permissions.

Verify source and Tool access independently

An Agent can discuss a file without being able to read it, and it can describe an operation without being able to perform it. Confirm source availability with a checkable question and confirm Tool availability through the appropriate interface or operator guidance. A role description should not be treated as evidence that every necessary capability is enabled. If the task requires a connected provider or external action, understand that boundary before supplying sensitive material or authorizing the action. Optional voice or correspondence-related configuration does not establish blanket offline functionality. Ask what the current deployment supports and what remains unverified. The useful result is a clear match between the role, the task, the available resources, and the permissions governing their use.

Evaluate the role with representative examples

Choose a few small examples drawn from the kind of work you expect to repeat. Include a straightforward task, a task with missing information, and a case that tests an important constraint. Define how you will assess usefulness before reading the responses. Look for factual accuracy, relevance, adherence to constraints, and appropriate handling of uncertainty. Keep the evaluation proportionate. You do not need a large benchmark to discover that an Agent consistently misunderstands the desired format. On the other hand, one successful example does not prove broad reliability. Record the result and adjust the task brief or consult the operator as appropriate. If configuration changes, repeat the relevant example rather than assuming the earlier result still applies.

Maintain the human review step

Specialization is helpful when it reduces the effort needed to begin a task or follow a useful process. It does not remove the need to inspect important facts, sources, and artifacts. Ask the Agent to explain its assumptions and identify what should be verified. For completed operations, examine the actual result instead of accepting the role's confidence as evidence. If the available Agent does not fit, you can still ask for a bounded explanation or a draft within its capabilities. Requests to create or modify Agent configuration should follow the controls and permissions established for the deployment. Early-access participants should work with their operator on those choices. The objective is a role you can understand and evaluate, not a label that encourages you to assume the application can do more than has been configured and verified.

Web guide: fridaylocalai.com/docs-agents

6. Skills & Tools

Know what guides an action.

A role, a procedure, and an executable action are different parts of an AI workflow. FridayLocalAI makes those distinctions explicit.

Agents define roles

An agent establishes a specialized purpose and associated configuration. Choosing an agent does not grant additional permissions or make its answers authoritative.

Skills describe procedures

Skills provide structured guidance for approaching work. Their lifecycle and availability are governed rather than inferred from a request alone.

Tools expose actions

A tool provides a defined operation. Server policy and the user's permissions determine whether an action can run; an instruction in conversation is not a substitute for authorization.

Review before relying

The Skills & Tools panel exposes governance information. Confirm that the intended capability is enabled and that its inputs, outputs, and boundaries fit the task.

Read the panel in context

Open the Dashboard and locate Skills & Tools where available. Resolved Skills describe guidance selected for the current context. Tools expose registered operations whose use is separately governed. Memory Notes are managed in the Memory panel. If an action is unavailable, request the appropriate operator review.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Explain the Skills & Tools information I can see: [non-sensitive labels]. Distinguish reusable guidance, tools that can act, and permissions. For my task, identify what is available and what still needs operator confirmation.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know

your current settings.

Begin with the action you need

When a task sounds like something the assistant should "do," separate the desired result from the mechanism. Explaining how to rename a file is different from renaming it. Drafting an email is different from sending it. Suggesting a sequence of steps is different from executing those steps against an actual system. This distinction helps you identify whether the task needs guidance, a Tool, or operator involvement. Describe the target and the expected outcome before asking for execution. If you only need an explanation, say so. If a real change is needed, confirm that the appropriate capability is configured and permitted. A confident response or an Agent's specialized name does not establish that it can perform the operation in your environment.

What a Skill contributes

A Skill provides reusable guidance for approaching a kind of work. It can describe a procedure, expected checks, or a way to organize a result. The Skills & Tools panel can show resolved guidance for the current context where available. The application's Skill handling is separate from the Memory Notes interface, even though both can influence how work is approached. Think of a Skill as an organized method rather than an unrestricted permission. A useful procedure might say how to review a source or prepare an artifact, but the actual operations still depend on available Tools and policy. If a step cannot run, the procedure does not authorize bypassing that limit. Ask what part can be completed and what requires a different approved capability or operator decision.

What a Tool contributes

A Tool exposes a defined operation through the runtime. Its inputs, target, permitted scope, and result matter. The existence of a registered Tool does not mean every user or Agent can call it for every purpose. Role, workspace, policy, and capability checks determine whether the requested operation can proceed. Before relying on a Tool result, understand what it proves. A lookup result can provide information; a file operation should produce a verifiable file state; an external action should have an appropriate confirmation or result. Do not stretch a narrow success into a claim that an entire workflow is complete. Ask Friday to separate the operation that ran from the remaining work and any uncertainty.

Read the panel in the selected context

Open Dashboard and inspect Skills & Tools where your role exposes it. Check that the intended project and conversation are active, because applicable guidance can depend on context. Read labels and status rather than assuming the panel is a shopping list of features that can all be enabled through a prompt. If something is unclear, describe the non-sensitive label to Friday and ask for an explanation. If a capability is missing, consult the operator about the intended configuration. Avoid making policy changes merely to get past an error. The useful outcome is a clear understanding of what the current task can do within the existing boundary and what additional review would be required.

Ask for a plan before a consequential operation

For a task involving changes, request a concise plan that identifies the target, the proposed action, and the result you will inspect. Include any boundary important to you, such as limiting work to a particular approved document or producing a draft without sending it. Clarify the point at which a separate decision is needed. After the operation, inspect the actual result. A plan and an attempted action are not completion evidence. If a result is partial, ask what succeeded and what remains unresolved. This is particularly important when a task mixes explanation, local data changes, and connected services. Treat each step according to what it does rather than assuming a single broad request resolves every authority question.

Use specialization without surrendering judgment

An Agent can help focus the assistant's role, and Skills can make recurring work more consistent. Those benefits do not remove the need to review factual claims, source use, and completed actions. A specialized assistant can still misunderstand the task or lack an important capability in a particular deployment. Start with a small representative exercise and evaluate its result. For recurring work, keep a clear description of inputs, expected outputs, and review criteria. Use Memory Notes for appropriate durable preferences and maintain Skills through the governed process available to the operator. The combination is most useful when guidance, permission, execution, and review remain understandable as separate parts of the workflow rather than being hidden behind a promise of general autonomy.

Web guide: fridaylocalai.com/skills-and-tools

7. Knowledge System

Sources & context

Stored is not the same as understood.

FridayLocalAI separates document handling, retrieval, and memory scope. Each has a different role in making information useful to a task.

Storage and readability

Confirm the file is accepted, then check whether its contents can be extracted or otherwise supplied in a supported form. An attachment badge alone does not prove readability.

Retrieval and task context

The relevant material must be selected and delivered to the task. Ask a narrow, verifiable question to check whether an answer actually uses the source.

Scoped memory

Memory Notes provide global, organization, project, folder, and conversation scopes. Do not assume every note or document belongs in every task.

Source authority

Keep provenance and revision in view. Generated summaries are aids to understanding; original evidence and operator authority remain controlling.

Make source use observable

Give the source a meaningful name and identify the section you want reviewed. Ask for a short quotation or precise passage reference, then check it against the original. If Friday cannot access the contents, supply a supported excerpt or ask your operator about processing. Do not ask it to fill gaps by guessing.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Use only the attached document if its contents are available to you. Identify the document, summarize the section about [topic], and point to the supporting passage. If the content is unavailable, say so and help me check the attachment workflow.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know your current settings.

Identify the question and its source

Knowledge work begins with a question and the material needed to answer it. A source may be a document, an excerpt, or information you type into the conversation. Identify the relevant source clearly and describe which part matters. Asking a narrow question first helps establish whether the response is grounded in the material you intended. For example, ask for the date under a particular heading or the difference between two stated requirements. Check the answer against the original. This is more informative than requesting a broad summary whose wording could have been generated from general background knowledge. Once source access is established for the relevant content, you can ask for a more substantial explanation or comparison with clearer expectations.

Storage is only one stage

An accepted upload means the application received material through a storage path. It does not, by itself, prove the model received readable contents. Different formats can require extraction, image handling, or another supported process. A preview that works in the browser may not establish the same access for the response workflow. Watch the available status and ask a checkable source question. If the content is unavailable, state that limitation rather than accepting a guessed answer. A small supported text sample can help isolate whether the problem is general or format-specific. Avoid repeatedly uploading private material without understanding the earlier result. Work with the operator when the source needs a different approved processing path.

Retrieval and relevance are separate questions

Material can be stored and readable without every part being relevant to every task. A useful response needs the appropriate context for the question. Identify the section, topic, or comparison you want rather than expecting an unlimited review of all stored information. Large or mixed source collections make it especially important to define the scope. Ask for references that let you inspect the supporting material. If the response cites the wrong section or omits a necessary source, correct the task context before continuing. Do not infer that the application has globally searched every document merely because an answer is confident. Context preparation is bounded by the available mechanisms and policies, and the user still needs evidence that the relevant material informed the result.

Memory serves a different purpose

Memory Notes can preserve guidance or information in an applicable scope, while documents remain sources to consult for their contents. A writing preference belongs to a different category of context from a technical specification. Keeping the distinction clear makes it easier to update guidance without confusing it with the original evidence behind a factual claim. Use the narrowest suitable scope for a note and verify the saved record. If a note summarizes a source, preserve enough provenance to understand its origin and limits. Do not allow a shortened memory entry to silently replace a detailed source when the distinction matters. A response may need both reusable guidance and a current source, and neither one grants permission to access unrelated information.

Review synthesis as well as quotation

A response can quote a passage accurately and still draw an unsupported conclusion from it. When asking Friday to compare or synthesize sources, request a distinction between what each source says, what follows directly, and what is interpretation. Check the important steps in the reasoning, especially when the sources differ or leave a question unanswered. For a practical exercise, provide two short descriptions and ask for agreements, differences, and unresolved questions. Review whether each comparison is supported. If one source is outdated, explain that context and ask how it affects the result. This develops a more useful knowledge workflow than treating every generated summary as a final authority simply because it includes a source reference.

Keep the workflow within its intended boundary

Source material can contain instructions written for another purpose. Those instructions do not automatically become commands from you to the application. State whether you want a summary, extraction, comparison, or explanation, and separate that request from the text being analyzed. An attached document is not permission to change files, contact services, or expand access. If the task involves proprietary information, confirm the processing and access arrangement with the operator. Private AI requires concrete handling decisions rather than assumptions based on a label. Preserve original sources and review important claims against them. FridayLocalAI is most useful as a knowledge workspace when it helps you organize questions, identify evidence, and make the remaining uncertainty visible instead of replacing that discipline with fluent but unverified text.

Web guide: fridaylocalai.com/docs-knowledge-system

8. Media Studio

Make space for creation.

Media Studio brings configured media workflows into Friday's Workshop through a dedicated panel.

Start with the enabled workflow

Check which generation options your deployment exposes. A visible interface does not imply every model, output format, or external service is configured.

Match the task to the environment

Generation workloads can have different model, storage, and compute needs from text conversation. Plan around the actual workflow and supported configuration.

Understand provider boundaries

Some capabilities may rely on configured external services. Review where inputs are sent and where outputs are stored before using sensitive material.

Review the result

Check the completed artifact and its suitability for use. A submitted job, a stored file, and an accepted final result are separate milestones.

Plan, generate, inspect

Open Media Studio from the workspace layout controls. Choose only a workflow enabled in your deployment. Describe the intended output, dimensions or duration where supported, and source material. Review the actual result for content, quality, and suitability before sharing it.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Help me prepare a brief for the Media Studio workflow I have available: [workflow]. Ask about purpose, audience, output, style, and source material. Explain what I should check after generation. Do not claim a job ran or a file exists until verified.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know your current settings.

Start with a creative brief

Before opening a generation workflow, describe the artifact you want to make and how it will be used. Identify the audience, subject, style, and practical output requirements. A small, clear brief is easier to evaluate than an open-ended request for something "amazing." If the workflow exposes dimensions, duration, or other settings, connect those choices to the intended use. Ask Friday to help refine the brief in a conversation. It can question unclear requirements and separate essential details from preferences. Make clear whether you want planning help or an actual generation attempt. A written prompt is not evidence that a job has been submitted, and a visible Media Studio panel is not proof that every possible media service is configured.

Check the available workflow

Open Media Studio using the workspace layout control. Inspect the options exposed by your deployment and role. The interface is intended for configured image, video, and audio workflows, but actual availability depends on services, models, providers, and operator setup. Choose the workflow that is both available and appropriate to the artifact you need. If an expected option is missing or unavailable, ask the operator rather than assuming a conversational instruction can enable it. Describe the intended result and what the interface shows. Keep this distinction especially clear when comparing a public description with an early-access installation: the description explains the product surface, while the actual configuration determines what can run in that environment.

Understand the input boundary

Reference material can help define a task, but it should be material you are permitted to use. Confirm the processing path before supplying sensitive or proprietary inputs. A local-first workspace may use a configured external provider for a particular media capability, so do not assume the same data path as a local text conversation. State how a reference should be used: preserve an approved layout, explain a subject, or guide a style choice. Do not let an ambiguous reference stand in for a complete brief. If the input cannot be read by the selected workflow, identify that limitation rather than allowing a guessed interpretation to proceed. The question is what the generation process actually receives, not merely what is stored in the workspace.

Make the first attempt reviewable

Use a bounded initial request that tests the essential idea. Decide in advance which characteristics matter most, such as the subject, composition, legibility, or relation to the brief. Avoid changing every setting at once after an unsatisfactory result, because that makes it harder to understand which change helped. When a job is submitted, inspect the available status and wait for the relevant completion evidence. A submitted request, a running job, a stored output, and an accepted final artifact are separate stages. If the process fails or stops, preserve the visible status before trying again. A retry should account for whether an output already exists and whether another attempt is appropriate for the configured service.

Review the artifact itself

Open the completed result and assess it against the brief. Do not rely only on a thumbnail, a filename, or a message saying the task is finished. Check content, dimensions or duration where relevant, quality, and suitability for the intended audience. If text appears inside an image or video, inspect the spelling and legibility directly. For revisions, describe the specific discrepancy and the desired change. "The title is unreadable at the intended display size" gives a clearer target than "make it better." Keep approved elements and requested changes explicit. Generated media may need further review before publication or sharing, and a successful technical output does not settle whether the artifact meets the creative or practical requirements of the task.

Preserve the useful result and its context

Keep the accepted output associated with a clear description of its purpose and the decisions that produced it. Use the storage and export workflow supported by your deployment. Verify that the file you intend to use is the one you actually open or download, especially when several revisions have similar names. If the result will be shared, perform the review appropriate to that use and confirm any relevant source permissions. Friday can help draft captions, a review checklist, or a concise revision brief, but it should not claim an artifact exists or was published without evidence. Media Studio is most useful when planning, configured execution, artifact inspection, and the final human decision remain distinct and understandable parts of the creative process.

Web guide: fridaylocalai.com/media-studio

9. Diagnose a Failed Response

Keep the evidence. Narrow the problem.

Start with a read-only investigation and the smallest reproducible task.

Record what happened

Capture the exact visible error, time, selected model, and steps. Keep secrets and private source contents out of support notes.

Check availability

Review application and model-service status without restarting services. Distinguish a sign-in problem from a failed model request.

Reduce the inputs

Try a short prompt without attachments, then add one file at a time. Observe whether the failure follows the input, selected model, or service.

Escalate with context

Use the approved support route with a concise reproduction. Preserve diagnostic evidence and do not modify stored data to make an error disappear.

Know when to stop retrying

First distinguish an incomplete response from a completed but incorrect answer. Note whether the message was saved and whether an artifact or tool action was already produced before retrying. If chat itself will not respond, capture the visible error and use the contact page instead.

Ask Friday to guide you

In an authorized FridayLocalAI conversation, try:

Help me investigate a failed FridayLocalAI response using this sanitized description: [details]. Ask about the visible status and whether anything was saved or produced. Suggest one safe check at a time and help me prepare a support report if needed.

Tell Friday what you can see and which step you have reached. If the instructions do not match your screen, say so; ask for the relevant Help topic or contact your operator. Friday cannot be assumed to see your screen or know your current settings.

Classify what happened

Before retrying, decide whether you saw no response, a partial response, a visible error, or a completed answer that did not meet the request. Those outcomes suggest different questions. Note whether your user message appears in the conversation and whether anything else was produced. A task involving an artifact or an external action may have progressed farther than the final text suggests. Record the time and the smallest set of steps that led to the problem. If an error message is visible, preserve its wording without including secrets or unrelated private content. Do not replace the observation with a diagnosis such as "out of memory" unless the available evidence actually establishes that cause. A precise symptom gives the operator a better starting point.

Check the selected context

Confirm the intended project, conversation, Agent, and Model controls. If a reply was produced, inspect its active attribution rather than relying only on the requested selection. A conversation opened through search may not be the one you expected, and changed context can affect what the task receives. If the issue concerns a source, confirm which file or excerpt was meant to be used. A filename displayed in the interface does not establish that the relevant contents reached the task. Ask a narrow source question if conversation is working. If it cannot answer from the source, investigate availability before treating the result as a model's failure to understand material it may never have received.

Try a bounded comparison

When appropriate, send one short, non-sensitive prompt without attachments. For example, ask for a three-item checklist based on a sentence you provide. Observe whether it completes. If it does, add one relevant input back to the task and compare the result. This can help identify whether the problem follows an attachment, a more complex request, or a particular workflow. Keep the comparison deliberate. Repeatedly sending the same long request can create more confusion without producing new evidence. If an operation might have already changed state, inspect the result before repeating it. The goal is to learn which condition matters, not to generate enough attempts that one happens to look successful while the underlying problem remains unknown.

Handle a wrong answer differently

If the reply completed but contains an error, identify the unsupported statement and compare it with the supplied material. Give a specific correction and ask for a revised answer. If the task was ambiguous, clarify the goal or missing fact. If the answer claimed an action occurred, inspect the corresponding state or artifact before accepting that claim. A poor response can come from unclear instructions, missing context, unsuitable capability, or an error in generation. Changing models may be one possible evaluation step, but it should not be the automatic response to every issue. First ask what evidence would distinguish those explanations. This keeps the investigation tied to the actual task instead of a vague search for a more impressive answer.

Escalate when the next step is operational

If the application, model service, storage workflow, or permission boundary needs investigation beyond the controls available to you, contact the operator. Do not alter system settings or restart services simply to see whether the symptom disappears. Preserve the current evidence and describe what you tried, including checks that succeeded. If chat is unavailable, use this website's guidance or the downloaded consumer documents to prepare the report. The assistant cannot be your only recovery path when the assistant itself is not responding. If the issue involves unrelated data becoming visible or an unexpected external action, stop the affected workflow and report the observation promptly without spreading the contents further.

Verify recovery using the original task

After an authorized repair or configuration correction, repeat a small version of the original failing workflow. Confirm the expected response, source use, saved state, or artifact. A generic "hello" response does not prove a document-processing issue is fixed. Match the verification to the symptom that led to the investigation. Record what passed and what remains uncertain. If the failure cannot be reproduced, say that rather than declaring the cause known. Keep a concise note of the outcome with the support exchange. Friday can help organize the observations, but the result you inspect is the evidence. A careful recovery leaves you with a clearer understanding of the workflow and a reliable next step if the same symptom returns.

A useful final check

Keep the original prompt and a sanitized account of the failure available for the operator. That makes it easier to compare the repaired workflow with the original symptom.

Web guide: fridaylocalai.com/help-diagnose-failed-response