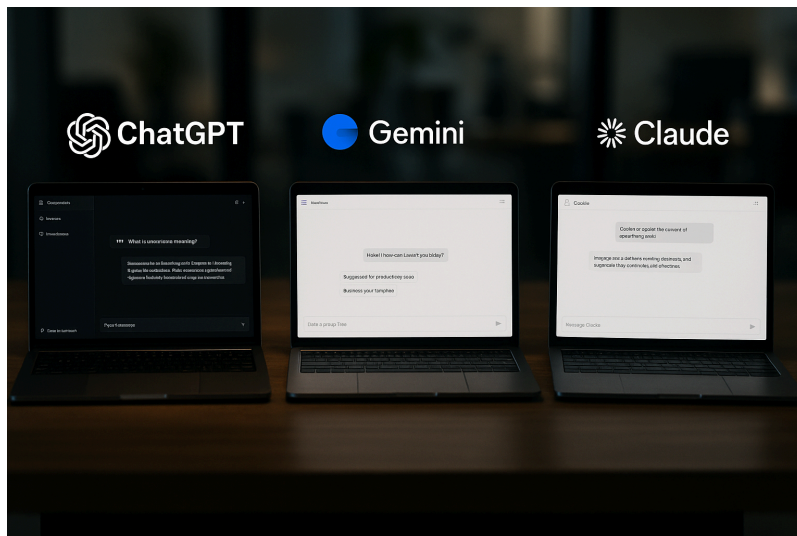


Comparing Conversational AI Tool User Interfaces 2025

By IntuitionLabs.ai • 6/5/2025 • 115 min read

conversational ai user interface ai tools ui design ai comparison chatgpt gemini claude productivity software usability



Comparing the User Interfaces of ChatGPT, Gemini, Claude, Poe, Llama, LeChat, and DeepSeek (June 2025)

Introduction

In the fast-moving AI landscape of 2025, business users have a variety of advanced conversational AI tools at their fingertips. Seven prominent options are OpenAI's **ChatGPT**, Google's **Gemini** (formerly Bard), [Anthropic's Claude](#), Quora's **Poe**, Meta's **Llama** (open-source implementations), Mistral AI's **LeChat**, and DeepSeek's AI assistant. While these systems all leverage [powerful language models](#), their **user interfaces (UI)** and features differ in ways that can significantly impact professional productivity. This article provides a detailed comparison of their UIs – from design and ease of use to collaboration, customization, and security features – to help business users determine which interface best aligns with their operational needs.

We will examine each product in depth and then present a comparison table summarizing key UI features across all seven tools.

ChatGPT (OpenAI) – The Polished Generalist

OpenAI's ChatGPT interface is often seen as a benchmark for clean design and usability. **Layout and Design:** The UI centers on a simple chat window with a collapsible sidebar listing conversation threads. It's a minimalist, distraction-free design that keeps the focus on the dialog. Users consistently praise ChatGPT's clean and organized layout – for example, its interface is noted as “*cleaner*” and more user-friendly than newer rivals like Mistral's LeChat [superannotate.com](#). The sidebar allows easy navigation between chats, and users can rename conversations to keep their prompt history organized [superannotate.com](#), a small but important feature for managing multiple ongoing discussions.

Ease of Use: ChatGPT requires virtually no training – open the app or website, type a prompt, and get an answer. The controls are intuitive, with clear buttons to regenerate replies or edit the last question if needed. On desktop, ChatGPT runs in the browser or via an official Windows app (introduced in late 2024) [techrepublic.com](#). On mobile, dedicated apps on iOS and Android provide a smooth experience consistent with the web version. The mobile apps even support voice input and voice replies, letting users “*ask questions out loud and have ChatGPT speak back*” [techrepublic.com](#) – a convenient option for hands-free use. The overall UX is polished and friendly for non-technical users.

Prompt Management & History: Chat histories are automatically saved (unless disabled), and appear in the sidebar for future reference. Users can search or scroll through past conversations (with recent chats pinned at the top) thanks to an infinite-scroll history pane introduced in late 2024 help.openai.com. Renaming chats is supported to help categorize them. Additionally, OpenAI's **"custom instructions"** feature allows users to set global preferences or context (like "assume my locale is UK" or specific tone guidelines) that apply to all new conversations. This persistent [custom prompt](#) acts like a personal system message, tailoring ChatGPT's behavior without repeated manual input. Business users find this helpful for maintaining consistency in style or incorporating company-specific context in every thread.

Customization & Extensions: ChatGPT is highly extensible through plugins and "GPTs" (custom chatbot personas). Plus subscribers can enable third-party plugins (e.g. for web browsing, databases, or business apps) right from the UI, integrating external tools into the chat experience. They can also switch between multiple models (GPT-3.5, GPT-4, etc.) via a menu at the top of the chat. In late 2023, OpenAI introduced a **"GPTs"** feature allowing users to create and share custom-tailored chatbots; these are essentially saved configurations with preset instructions and optional training examples. The interface provides a simple form to define a custom bot and even a community **GPT store** for discovering ones others have made. While still evolving, this offers a no-code way to customize ChatGPT for specific roles (e.g. a "Travel Advisor" GPT).

Integration with Business Tools: Natively, ChatGPT is a standalone web/app experience and doesn't directly plug into business software out-of-the-box. However, OpenAI offers an API for developers to integrate the ChatGPT models into their own applications or workflows. Many companies have built custom interfaces on top of ChatGPT's API to connect it with internal knowledge bases or enterprise software. For end users, ChatGPT Enterprise (launched August 2023) and the newer ChatGPT Team plan provide some integration-friendly features. The **ChatGPT Windows app** can hook into developer IDEs like VS Code and even the command line, allowing developers to query ChatGPT contextually from their coding environment techrepublic.com. Additionally, ChatGPT can handle files and data through its **Advanced Data Analysis** tool (formerly "Code Interpreter"), letting users upload spreadsheets, PDFs, or datasets for analysis within the chat. This acts as a mini integration: for example, a financial analyst can upload a CSV and ask ChatGPT to generate pivot tables or charts, with the interface providing file import and Python execution capabilities. While not a direct plug-in to business systems, it significantly extends ChatGPT's utility for data-heavy tasks.

Collaboration & Workspace Support: ChatGPT's core interface is single-user – one user per conversation. There's no live multi-user editing of a chat. However, ChatGPT Enterprise and Team address collaboration in other ways. **ChatGPT Team** (targeted at small organizations) introduces a *"secure, collaborative workspace"* for multiple users openai.com. In practice, this means an organization can have an admin-managed ChatGPT environment where members share access to company-provided chat credits and features. Team admins get a console for user management (with SSO support and domain verification) adventuresinacre.com. Users within

a team can easily share conversation links with each other, and the team “*collaborative workspace*” promises that shared chats remain internal and private [adventuresinre.com](#). (It’s worth noting that *collaboration* here is mostly in sharing outputs; ChatGPT doesn’t yet have true real-time co-authoring of a single chat by multiple people the way Google Docs allows simultaneous editing.) Still, being able to share chats or use common custom GPTs across a team is valuable for consistency. Enterprise users can also create **shared chat templates** – for instance, a pre-defined prompt workflow that colleagues can all use – which streamlines repeated business processes. Overall, ChatGPT’s team offerings are evolving towards treating the AI assistant as a communal productivity tool, but direct multi-user dialogues with the AI are not the norm (each user initiates their own chats).

Accessibility & Multimodal Features: ChatGPT supports multiple modes of interaction, which is great for accessibility. The interface offers **text-to-speech** and **speech-to-text** on mobile – one can press a button and speak a query, and ChatGPT will transcribe it and even *reply in a synthetic voice* [techrepublic.com](#). This voice mode makes the assistant more accessible to users with visual impairments or those on the go. Moreover, ChatGPT accepts **image inputs** on all platforms (Plus users can upload images for GPT-4 to analyze, thanks to its vision capabilities [techrepublic.com](#)) and can generate images via the DALL-E 3 model. In fact, as of early 2024, all ChatGPT users can generate images in-chat, with AI-generated content automatically watermarked for transparency [techrepublic.com](#). The UI for image upload/generation is seamlessly integrated: you can attach an image and ask questions about it, or request the AI to create an image from a prompt, all within the conversation flow. Additionally, OpenAI has introduced a **Canvas** feature (in late 2024) – an experimental visual whiteboard within ChatGPT for organizing text, code, and images spatially [help.openai.com](#) [help.openai.com](#). This is useful for visual thinkers or brainstorming sessions, though it’s still a secondary interface. In terms of language accessibility, ChatGPT can converse in many languages and the UI is available in multiple localizations. Business users across the globe leverage ChatGPT for translation and multilingual support (though rival Gemini has an edge in certain languages, as we’ll see).

Handling Long Inputs, Code, and Tables: ChatGPT (especially GPT-4) is adept at working with lengthy documents and code. GPT-4 offers an **8K token context by default, with a 32K option** for Enterprise users [adventuresinre.com](#), which means it can ingest around 50 pages of text in one go. In practical terms, users can paste large reports or contract documents and get summaries or analysis. The interface doesn’t visibly show a “token meter,” but it will warn if an input is too large. Still, compared to Claude’s 100K context, ChatGPT may require breaking up very large files. To mitigate this, the Advanced Data Analysis tool allows uploading files (up to dozens of MBs) that the AI can read entirely, enabling work with large datasets or long transcripts beyond the raw context window. When it comes to **code**, the ChatGPT UI formats any code output in clear monospaced blocks with syntax highlighting, and includes a handy “*copy code*” button on each block for user convenience. The Code Interpreter mode goes further: the UI opens a sandbox where code can be executed and results (including charts or files) can be returned within the chat. This makes ChatGPT a powerful assistant for programmers and analysts, directly within the chat interface. For **tabular data**, ChatGPT can output Markdown

tables which render as nicely formatted tables in the UI – useful for summaries or comparisons (like the table we'll present below). In short, the interface is well-suited for technical content, and OpenAI's updates continually improve how such content is displayed and manipulated.

Security & Privacy Controls: With enterprise adoption in mind, ChatGPT's interface and policies include robust privacy features. All users have the option to turn off chat history (and thus model training on their data) with a single toggle in settings, which ensures no conversation content is retained on OpenAI's servers beyond 30 days (and even then only for abuse monitoring) adventuresinai.com. ChatGPT Enterprise and Team go further by default – OpenAI guarantees that **no conversations on those plans are used to train models** adventuresinai.com, and all data is encrypted in transit and at rest (SOC 2 compliance). The Enterprise UI also provides an **admin panel** where company admins can manage seats, enforce two-factor auth, and monitor usage. While these controls are largely outside the chat interface itself, the user-facing UI does display reassuring cues (for example, a message about data usage policy on first login, and a lock icon indicating a verified corporate domain account). OpenAI's commitment to privacy is a selling point for companies: *"all conversations are encrypted... and not used to train OpenAI's models"* adventuresinai.com. For end users, the ability to delete chat history or export it for auditing is easily accessible. Overall, ChatGPT marries a consumer-simple interface with enterprise-grade security options – a key reason it remains a top choice for many business users superannotate.com.

Desktop vs Mobile Experience: ChatGPT offers a unified experience across platforms. On a desktop browser, users benefit from a roomy interface where sidebars can be pinned open for quick context switching help.openai.com. There is also an official ChatGPT desktop app for Windows (with Mac likely in development) that behaves similarly to the web UI but allows a global shortcut and better OS integration techrepublic.com. On mobile, the ChatGPT app adapts the UI to smaller screens with a collapsible menu for history and a microphone button for voice input. Feature-wise, mobile got some capabilities even before web – for example, **real-time voice conversations with screen sharing** rolled out on mobile in late 2024 techrepublic.com. Users could show their phone screen or camera view to ChatGPT and talk through a problem, which is quite innovative for on-the-go support (e.g., getting help debugging code by sharing a snippet on your screen via the app). Both iOS and Android apps support image uploads (e.g. snapping a photo of a diagram or data table to ask questions about it). Conversations sync between devices when logged into the same account, so you can seamlessly switch from your laptop to phone. The mobile UI is optimized for touch, with large microphone and image buttons, making it very accessible. In summary, ChatGPT provides a **consistent, rich experience on desktop and mobile**, ensuring users have AI assistance wherever they work.

Google Gemini – The Integrated Multimodal Assistant

Google's **Gemini** represents a fusion of its earlier Bard chatbot with the company's powerful new Gemini AI models and the Duet AI workplace assistant features. By June 2025, **"Gemini"** isn't

just a model name but a suite of AI capabilities accessible through various Google interfaces.

Layout and Design: The primary way business users interact with Gemini is via the **Gemini conversational app**, which is analogous to ChatGPT's interface but deeply integrated with Google's ecosystem techcrunch.com. On the web, Gemini lives at a dedicated site (gemini.google.com) and presents a familiar chat layout: a dialog pane with the AI's responses and a text input box. However, the design language follows Google's Material aesthetic – a clean white background, Google Sans text, and blue accent on the AI's replies. A unique aspect is how **Gemini's interface is omnipresent across Google products**: there's a Gemini side panel in Gmail and Docs, an overlay on Android, and hooks in many Google apps. For instance, on Android devices, pressing and holding the power button or saying "Hey Google" now invokes a **Gemini overlay** that can answer questions about whatever app is currently on screen techcrunch.com. This is a novel UI paradigm – instead of a static chat window, Gemini can appear contextually (like on top of a YouTube video or a Chrome page) to provide relevant help. In Google Workspace apps, the Gemini UI often appears as a sidebar assistant. For example, in Google Docs a panel labeled "Help me write" is essentially a Gemini chat specialized for document editing. The **design across these contexts is consistent** – Gemini's responses are shown with options to refine or regenerate, and suggestions to use outputs (like "Insert" into document or "Send" in Gmail).

Despite this flexibility, Google ensures a cohesive user experience by syncing conversations and preferences across all instances. Whether you chat with Gemini on your phone or your laptop, you can access the same conversation history when signed in with your Google account techcrunch.com. This cross-platform continuity is a strong suit of Gemini's UI integration.

Ease of Use: Gemini is designed for broad accessibility, leveraging Google's familiarity. If you've used Google Assistant or Bard before, interacting with Gemini feels natural. The entry point is often right where you work – need an email drafted? Click the prompt in Gmail's UI; have a question while browsing? Open the overlay. This reduces friction since you don't always have to navigate to a separate chat app. The interface itself uses friendly language and even provides **suggested follow-up questions** to guide users (a feature carried over from Bard). One thing to note: earlier versions of Bard did not save conversation context by default, but with Gemini's introduction, Google added a **Memory feature** for Gemini Advanced users, which stores past conversation context and user preferences for more personalized replies techcrunch.com. This means Gemini can *remember* things you told it earlier (e.g., your project details or style preferences) and bring them into new responses – boosting ease of use by reducing repeated explanations. Users can manage this memory in settings, including wiping it if needed.

The UI also includes quality-of-life features like **draft answer variants**. Similar to Bard's original "View other drafts" function, Gemini often generates multiple approaches to a response. The interface might show one answer, but allow the user to click and see two alternative phrasings or solutions if they want a different style – great for business users looking for the best wording or idea. There's also a **"Google It" button** (a small G icon) that lets users quickly fact-check the

AI's answer with a web search en.wikipedia.org. This appears after responses, integrating the power of Google Search for verification.

Prompt Management & History: All your interactions with Gemini (across devices) are tied to your Google account. In the standalone Gemini web app, a sidebar shows recent conversations, similar to ChatGPT's interface, and you can name or revisit them. Google also added a search bar for your chats – searching your past Gemini conversations by keyword (a feature reportedly under development to make finding that code snippet or answer from last week easier) reddit.com. As mentioned, **Gemini Advanced** (the premium tier) introduced a **persistent memory**. This goes beyond simple chat history; it allows the model to recall facts you provided in earlier sessions. For example, a user could remind Gemini of their team's OKR goals in one chat, and weeks later Gemini might proactively use that context when asked for project advice. This is akin to custom instructions or a long-term memory vault. Business users appreciate this as it makes the AI feel more personalized and eliminates retyping context.

Custom prompt management is also facilitated through **"Gems"** – Gemini's version of custom chatbots. Advanced users can create what are called **Gem** templates by describing a role or task for a chatbot, e.g. *"You are my agile project management coach"*. The system will then generate a custom chatbot with that persona, which you can refine and even share with colleagues techcrunch.com. These Gems act like saved prompt+context configurations, accessible on both desktop and mobile. In the Gemini app, there's a section to manage Gems, and they can be pinned for quick reuse. This feature parallels ChatGPT's custom GPTs and Poe's user-made bots, enabling prompt reusability and specialization through the UI.

Customization Options: Beyond custom Gems, Gemini offers a range of settings to tailor the experience. Users can choose the **mode or model size** if they have access – for instance, selecting "Gemini Ultra 1.0" (the most powerful, if subscribed to Google's AI Premium) versus the standard model. In fact, Google One's AI Premium Plan unlocks **Gemini Advanced**, which not only uses the larger Ultra model but also adds toggles for things like **"Deep Research"** mode techcrunch.com. Deep Research is an experimental feature in Gemini 2.5 Pro that, when activated, makes the AI take extra steps: it *"develops a multi-step research plan and searches the web"* to produce a thorough answer techcrunch.com. This is essentially a built-in tool usage/chain-of-thought mode – the UI shows Gemini thinking in stages, perhaps analogous to DeepSeek's approach, though Google keeps the internal reasoning hidden and just presents the final compiled answer with citations. Users can toggle this on for complex queries (e.g., market analysis) and off for simple ones, trading speed for depth. Additionally, Gemini Advanced users can **run Python code within the chat** techcrunch.com (similar to ChatGPT's Code Interpreter). When this feature is used, the UI provides a code editor pane and executes code, showing outputs or errors – extremely useful for data analysis tasks. All these options are neatly presented in the interface, often under a settings menu or as prompts like "Use Advanced mode for this query." Google has struck a balance between simplicity for casual use and powerful options for those who need them.

Integration with Business Tools and APIs: If there's one area Gemini shines, it's integration. Gemini is essentially woven into Google's productivity suite and beyond. In Gmail and Google Calendar, Gemini (via Duet AI branding in those contexts) can read your emails or agenda (with permission) and perform actions like drafting responses or summarizing threads techcrunch.com. The UI for this is a side panel or context menu within Gmail – you might see a “Help me respond” button on an email thread, which opens the Gemini assistant to draft a reply using the thread's content. In Google Docs, a “Help me write” or “Help me brainstorm” side panel leverages Gemini to generate content or outline documents, directly inserting text into your doc on command techcrunch.com. In Sheets, Gemini can be asked (via a sidebar prompt) to create formulas or even analyze data in the sheet, producing tables or charts. There's also integration in Google Slides (suggesting images or generating slide content), and even Google Meet (where Duet AI can attend meetings, take notes, and provide summaries). The interface in Meet, for instance, might show a live transcript with AI highlights or an automatic summary at the end for all participants.

Crucially, all these are *native UI integrations* – meaning business users don't have to leave their workflow to use Gemini's AI. The AI Premium Plan ties it all together, giving companies the ability to deploy Gemini across Workspace apps techcrunch.com with administrative controls. From an API standpoint, Google Cloud offers the **Gemini API (Vertex AI)** for developers to incorporate the models into custom apps ai.google.dev. But many businesses may find they need no custom dev work – the out-of-the-box integrations cover email, documents, spreadsheets, and more. There is also **Gemini in Google Maps** for analyzing local business info techcrunch.com, and even in Google's Chrome browser (an experimental “Ask Google” feature in Chrome can have Gemini summarize web pages or answer questions about them). In essence, **Gemini acts as a ubiquitous assistant across the Google ecosystem**, which is a huge advantage for companies already using Workspace. This tight integration is something neither OpenAI nor Anthropic (which rely on third-party plugins or APIs) currently match at the same scale.

Multi-User Collaboration & Workspaces: While Gemini doesn't have a “team chat” feature in the style of Slack, its integration in collaborative Google apps means the AI can naturally participate in multi-user workflows. Consider Google Docs: multiple users can be editing a doc simultaneously, and the Gemini-powered assistant can be invoked by any of them. The suggestions it provides (say, rewriting a paragraph) are visible to all collaborators, and inserting AI-generated text is just like another collaborator contributing. This is a form of **AI-in-the-loop collaboration** – the UI doesn't distinguish AI suggestions from normal user actions in the revision history, so it seamlessly fits into team editing sessions. Another example is Google Meet: if AI summaries or action items are generated for a meeting, all attendees receive them. This is a collaborative output of the AI used in a multi-user context.

That said, Gemini's standalone chat app is generally single-user (tied to your personal account). But Google does allow sharing of Gemini conversation transcripts through links – for instance, you can click “Share” on a Gemini chat and get a link (much like Bard had en.wikipedia.org), which colleagues can open to view the conversation (read-only). Google has implemented

safeguards here: a privacy incident in 2023 involved shared Bard conversation links being indexed by Google Search accidentally en.wikipedia.org, which was quickly fixed. Now, sharing a Gemini chat is deliberate and explicit, and enterprise admins can likely disable it if desired. For internal use, Google's approach is that every user has their own AI assistant, but all those assistants can tap into *shared company data* if integrated (for example, a company might connect Gemini to an internal knowledge base via Google Cloud Search). So while you don't "chat together" with Gemini, you benefit from a common pool of information and the same AI capabilities in a group setting.

In terms of workspace management, Google Workspace admin console now includes controls for Duet/Gemini – e.g. an admin can turn on/off AI access for certain departments, set data usage policies (Google ensures that **Workspace customer data is not used to train** Gemini models unless companies opt-in) and review logs of AI activities for compliance. This gives enterprises confidence to deploy Gemini widely. The UI impact for end-users is mainly that they might see a notice like "Your organization enables AI assistance. Your interactions won't be used for model training" – similar to ChatGPT Enterprise's assurances.

Accessibility Features: Google has a strong track record in accessibility, and Gemini's multi-modal nature enhances that. **Voice support** is built-in: on mobile, users can talk to Gemini using the familiar "Hey Google" hotword or a mic button, and Gemini will respond with spoken output in a natural voice venturebeat.com. In fact, Gemini's voice mode on mobile one-ups ChatGPT by integrating with personal data (with permission): a user could verbally ask "What's next on my calendar?" and Gemini will check Google Calendar and speak the answer venturebeat.com. This voice interface supports multiple voice personas; Google offers choices (names like "Mellow" or "Glassy" voice, etc.) to suit user preference venturebeat.com.

For users with visual impairments, having a full voice assistant that's actually backed by a powerful LLM is game-changing – it can describe images if you upload one, or read out documents from Drive. Speaking of images, **Gemini is multimodal**: you can upload images or PDFs to the chat and ask questions about them techcrunch.com. The UI has an attachment button (paperclip icon) to add files from your device or directly from Google Drive. For example, a user could upload a lengthy PDF report; Gemini can summarize it or answer queries, leveraging Google's **NotebookLM** tech to even turn documents into conversational sources techcrunch.com. By February 2025, Gemini could handle audio and even video input to some extent: e.g., you might provide an audio clip for transcription or ask Gemini to analyze the content of a video (though video understanding is in early stages, Google did demonstrate Gemini generating interactive charts from data and even some **video generation (Veo 3)** for premium users blog.google).

Language accessibility is a major plus: Google's AI supports **over 40 languages**, and I/O 2025 announcements highlighted improved multilingual support and even emotional tone handling in voice timesofindia.indiatimes.com. Business users in non-English markets often find Gemini more adept at their local language and culturally aware, given Google's training data breadth.

Additionally, for hearing-impaired users, Gemini's integration in Meet can live-caption and summarize spoken conversations, acting as an assistive agent. It's clear Google is pushing Gemini as an AI for *everyone*, with robust accessibility and multimodal interfacing built in.

Support for Long Documents, Code, and Data: Gemini's capability here is strong and improving rapidly. The **Gemini Ultra** model variant is designed for heavy-duty analysis – Google has noted it can *"tackle difficult problems, analyze large databases, and more"* ai.google.dev. In practice, a user can paste or attach very large texts (dozens of pages) for analysis. Google hasn't publicly stated a token limit, but anecdotal usage suggests Gemini can comfortably handle around 30k tokens of input on Advanced (and this may increase as newer versions come out). Moreover, because Gemini is integrated with Google Drive and cloud storage, you don't always have to paste content – you can point it to a file. For example, *"Summarize the Q4_report.pdf in my Drive"* will have it fetch the file (with your permission) and process it. This bypasses some context window limitations by streaming content as needed.

For **coding**, Gemini provides a competitive experience. In the standalone chat, code outputs are formatted in proper code blocks with syntax highlighting. Google's integration with developers' tools is also notable: Gemini can be accessed in Colab notebooks or via the Cloud console, meaning developers can chat with it alongside their code. Gemini Advanced even allows executing Python code within the chat techcrunch.com. This is presented in the UI with a dual pane: left side for code (with line numbers, etc.) and right side showing output or errors. It essentially replicates what ChatGPT's Code Interpreter does, but tied into Google's ecosystem (potentially running on Google's cloud for heavy computations). For data analysis, Gemini's responses can include formatted tables (it will use Markdown tables or embedded HTML tables in the chat UI to neatly present rows and columns of data). In Google Sheets, asking Gemini something like *"Sort this data by region and total sales"* can result in it actually manipulating the sheet or producing a new sheet/tab with the sorted table – a very direct action on data via the UI. This kind of deep integration with data is unique to Google's approach.

Security & Privacy: Given Google's business user base, Gemini's interfaces incorporate strong privacy and security measures. At launch, Google made it clear that interactions within **Google Workspace (enterprise accounts)** would *not* be used to train the AI models en.wikipedia.org. The UI in enterprise Google products often displays a shield icon or a small disclaimer "Duet AI (Gemini) does not use your content for model training." All data stays within the company's Google Cloud instance, benefiting from Google's encryption and security compliance (Google Cloud meets stringent standards like ISO 27001, SOC, etc.). For example, if a corporation enables Gemini help in Google Docs for its employees, those document contents analyzed by AI are not leaving their trusted cloud boundary or being seen by Google staff – it's processed and discarded. Admins can set retention policies for AI interaction logs, or disable the memory feature if they don't want any cross-session data persistence. There are also user-facing controls: you can clear your Gemini chat history, and even in your Google Account settings there is now an "AI data" section where you can manage what info is stored. Google's **AI Principles** mean the UI is filled with reminders about responsible use – every Gemini response

still carries a disclaimer that it may be inaccurate, and sensitive queries trigger warning banners (just as Bard did).

From a security perspective, the integration with Google's identity and access management is a boon. If you're logged into a managed Google Account, Gemini respects all your organization's security policies (2FA, data loss prevention rules, etc.). For instance, if a company prohibits external sharing, Gemini's share chat link feature will be disabled for those users. IT departments can monitor usage of Gemini through Google's admin dashboards – providing transparency if needed (like seeing how often employees use it, though not the content of queries, preserving privacy). Another aspect is **Google-Extended**, an initiative allowing websites to opt out of being scraped by Gemini en.wikipedia.org. While this is more about the training side, it reflects Google's attempts to balance AI reach with privacy.

Overall, **Gemini's UI aims to be enterprise-ready by default**, embedding trust indicators and giving organizations control. The trade-off for this security is that outside of Workspace, free consumer use of Gemini may have some data usage (like model improvement) unless you opt out. However, Google has strongly aligned Gemini's professional usage with strict privacy, knowing that's the expectation set by competitors like OpenAI's enterprise plan.

Desktop vs Mobile Experience: Gemini is truly cross-platform but not in the same standalone way as ChatGPT. On **desktop**, you might access Gemini in multiple forms: via the web interface (gemini.google.com) for a full-screen chat experience, or within other Google web apps (Docs, Gmail, etc.). The dedicated web interface is relatively new (launched in early 2024 when Bard was rebranded), and it includes features like a collapsible sidebar for history and Gems, much like ChatGPT's. If you have a Google One Premium subscription, a badge indicates you're using Gemini Ultra for better quality. It's a smooth experience and runs in modern browsers without hassle.

On **mobile**, there isn't a single "Gemini app" on iOS in the traditional sense; instead, the **Google app** and **Google Search app** on iOS serve as the Gemini client techcrunch.com. You open the Google app and tap the chatbot icon or issue a voice command to start chatting with Gemini. On **Android**, Google actually replaced the old Assistant app with a new **Gemini app** techcrunch.com, which can be launched by voice or a long-press of the power button. This Gemini app on Android essentially merges the classic personal assistant with the new chat model. It can do all the phone-specific tasks (opening apps, setting reminders) and also open a full chat for general queries. The UI feels like a superset of Google Assistant and Bard – you have voice or typing input, and visual cards for certain answers (e.g. if you ask Gemini for today's weather, it still shows a small weather card). Conversations on mobile can be scrolled back and also appear on desktop if you switch – the sync is instant via your account techcrunch.com.

One standout on mobile is how **context-aware** Gemini is: ask it a question while viewing an article on your phone, and it can take into account the article text (with your permission) to answer or summarize – all without manual copy-paste. This is thanks to Android's intents and overlay – a very convenient UI trick for busy professionals reading on mobile.

Google has also been rolling out **Gemini Live** on Pixel devices en.wikipedia.org – essentially an always-on top-of-screen prompt that you can use even when the phone is locked (for quick questions). This is similar to iOS's Siri suggestions, but powered by the more capable model.

In summary, the Gemini experience is **pervasive and context-driven on mobile**, and richly integrated but somewhat fragmented on desktop (since you encounter it in different apps). Users benefit from not needing to switch contexts – the AI comes to you. For companies heavily in the Google ecosystem, this means employees get AI assistance within tools they already use daily, on both their workstations and smartphones, with a very low barrier to entry.

Claude (Anthropic) – The Conversational Colleague

Anthropic's **Claude** has carved a niche as a large-context, friendly AI assistant, and its interface reflects an emphasis on lengthy, in-depth conversations and team productivity. **Layout and Design:** Claude's main interface (accessible via [Claude.ai](https://claude.ai) on the web and through its mobile apps) is a chat screen with a minimalist design similar to ChatGPT, but with its own subtle style. Claude uses two columns: the left sidebar for a list of conversations and "Projects" (more on those shortly), and the main area for the chat. The design is clean and utilitarian – black text on a white background (or dark mode if chosen) with purple accents for the Claude logo or loading spinner. Each message is labeled with either "You" or "Claude" and time-stamped. The interface deliberately downplays itself to let the content shine, a philosophy Anthropic has due to their safety-first approach (they often include system messages about not revealing policies, etc., but those are hidden from the UI unless a policy issue arises, in which case Claude might explain it in the conversation).

One signature design element is Claude's **ability to handle extremely long inputs and outputs** – up to 100,000 tokens in Claude 2 and even more (200K) in the newer Claude 3.5 "Sonnet" model pymnts.com. The UI accommodates this by allowing very long scrollback. To keep navigation manageable, Claude's web interface added features like an **outline view for long responses** – for example, if Claude produces a 20-page report, the top of the message may have a generated outline or summary that you can click to jump to sections. This is an innovative UI solution to lengthy AI output, ensuring users aren't overwhelmed by walls of text.

Ease of Use: Claude is designed to feel like a helpful colleague in chat form. It often signs off answers with a polite prompt like, "Let me know if you have any questions about this!" to encourage further dialogue. Using Claude is straightforward: you type in the text box and hit enter. There's no complex setup for basic use. Where some complexity comes in is **Claude's advanced features** like Projects and Artifacts (targeted at power users), but Anthropic has kept these tucked away unless you need them. Casual users can ignore Projects altogether and just use Claude like any other chatbot. For those who engage with the deeper features, the interface provides tooltips and gentle onboarding. For instance, the first time you create a Project, Claude

shows a brief explanation: “Projects let you organize your chats and knowledge” and suggests example uses.

Claude also tends to be **more conversational and less terse** than some AI, which users either love or find verbose. Anthropic tuned Claude to be extremely polite and comprehensive. If brevity is needed, users can instruct it accordingly. The UI has an easy-to-access **“Tone and Length”** setting when starting a new conversation or project – a small dropdown where you can choose preferences like “Formal” vs “Casual” tone, or “Short” vs “Detailed” answers. This is a nice quality-of-life feature for business users who may want, say, a very formal tone for client-facing content or a concise answer when in a hurry.

Prompt Management & History: In Claude’s interface, every conversation is saved in the sidebar, and you can rename them at will. Anthropic went a step further in mid-2024 by introducing **Projects**, which are essentially folders or workspaces for chats pymnts.com. A Project can contain multiple chat threads as well as a set of reference documents or data, acting like a mini knowledge base. This is a unique approach to managing prompts and context. For example, a legal team might create a “Contracts Project” in Claude, upload several policy documents (as Artifacts), and then have multiple chat conversations in that project analyzing or drafting contracts, all drawing on the shared reference docs. The UI shows Projects in the sidebar above individual conversations. Clicking a Project opens a view with the list of all chats under it and any uploaded documents. This structure makes it much easier to organize work by topic or team, rather than dealing with one long, monolithic chat or dozens of separate chats with no grouping.

Within each Project or chat, Anthropic recently added **Custom Instructions** (similar to OpenAI’s feature) pymnts.com. You can set per-Project guidelines that influence Claude’s responses – e.g. “Use a friendly tone and always answer in Spanish in this project.” These instructions persist for all chats in that Project, so you don’t have to repeat them. This is extremely useful for business use-cases like setting a brand voice or factual baseline.

Claude’s history handling also benefits from its long memory – you can usually refer back to something said many turns ago without the conversation context dropping, given the large token window. However, to help manage this, Projects allow *resetting* context without losing the thread of discussion – one can start a fresh chat within the project if the previous chat became too context-heavy or off track. All chats in a Project still have access to the Project’s uploaded knowledge and custom instructions, which is an elegant way to maintain continuity of knowledge without carrying forward irrelevant conversational clutter. Users have praised this ability to “start fresh but informed” as making Claude feel more flexible for extended research tasks pymnts.com.

Customization Options: Claude’s personality can be customized implicitly by user instructions, and Anthropic also provides explicit controls for certain behaviors. While it doesn’t have user-created plugin capabilities, it does allow **setting the conversation mode**. Aside from tone/length as mentioned, Claude has modes like “Balanced”, “Precise”, or “Creative” which

adjust how adventurous vs strictly factual it should be – similar to how Bing Chat offered different modes. Business users often keep Claude in a balanced or precise mode to ensure reliability.

Anthropic's Claude Pro and Team plans unlock further customization in the form of **Artifacts and extended settings** pymnts.com. **Artifacts** are a feature where Claude can output content into different formats or panels. For instance, if you ask Claude to draft a Python script, you can open an Artifact panel where the code is displayed in a full-size editor separate from the main conversation, making it easier to review or copy pymnts.com. If you request a diagram or a small web page layout, Claude can generate it as an Artifact that you open in a side window (with HTML rendering, etc.). This is effectively an in-chat *file generation* or *preview* ability. It's cited to be *"particularly useful for developers, offering a larger code window and live previews for front-end work."* pymnts.com. The UI makes it simple – a generated Artifact appears as a clickable card in the chat, which when clicked, expands alongside the chat.

Additionally, Anthropic is exploring **Memory features** (beyond context window) where Claude can remember user preferences over time globally (Anthropic staff hinted at this on forums reddit.com). Already, Claude Team offers an account-level setting where you can input some general preferences or facts (like "Our company sells X, assume that context"), which it will apply to new conversations by default. This is analogous to OpenAI's custom instructions but aimed at organizational use.

Integration with Business Tools and APIs: While not as deeply embedded as Gemini, Claude has been integrating with popular business platforms. Anthropic partnered with tools like **Slack** and **Notion** to allow Claude's usage within those apps. For example, there's an official Claude Slack bot – employees can add Claude to a Slack channel and converse with it there, using Slack commands. The interface in Slack is essentially the same chat but happening in a Slack window, making it handy for teams to collectively query Claude (e.g. summarizing a Slack thread or brainstorming ideas right in the channel). Similarly, Notion has an AI assistant feature, and Anthropic is one of the model providers behind it. The Notion UI then offers Claude's capabilities when you press Space and ask AI to draft or summarize content in a Notion page.

Anthropic has also been integrated into services via **API and platform deals**. Through **AWS Bedrock**, Claude is available to enterprises to plug into their software aws.amazon.com. And Claude's API can be used by developers much like OpenAI's, so businesses have built custom internal tools (for instance, a financial analysis dashboard with Claude answering questions about company data).

One notable integration is **Claude Voice** and **Claude 4's planned browser tools**. In mid-2024, Anthropic launched **Claude voice mode on mobile**, which allowed Claude's app to connect to certain personal data like Google Calendar, Gmail, and Google Drive (with user permission) venturebeat.com venturebeat.com. This was a direct integration aimed at rivaling Google's assistant capabilities. Using just your voice, you could ask Claude "Read my next meeting from Google Calendar" or "Summarize the Q1 Sales spreadsheet from Drive" and it would fetch that

info (for Pro subscribers). The UI flow involves linking your Google account in Claude's app settings once, then using natural voice commands. This was a bold integration and well-received by power users, though it's gated behind the paid tiers (Claude Pro at \$20/mo and Claude Max at \$100/mo got these features) venturebeat.com.

Claude's web search integration is also worth noting: initially it lacked web access, but by 2024 Anthropic enabled an **in-chat web search** for all users (even free) venturebeat.com. A user can toggle a "Search the web" option before sending a query, and Claude will fetch live information from the internet. The UI indicates this by showing sources and citations in the response – for example, it might list footnotes linking to webpages it referenced (much like Bing or Perplexity AI). This keeps Claude's answers up-to-date and is very useful for market research or news queries. Unlike ChatGPT's plugin, it's built-in and one-click to use, enhancing integration with the broader web.

Though not a traditional "business tool," it's notable that Claude has a **browser extension** that lets you select text on any webpage and summon Claude to explain or summarize it. This bridges the gap between the Claude web app and everyday browsing or web-based SaaS tools a user might be using.

In summary, while Claude's native app isn't embedded in enterprise software by default, **Anthropic has made Claude available in the environments where businesses work** – Slack for communication, Notion for documentation, Google services via voice, and general web browsing. Companies that want deeper integration can use the Claude API or self-host via partnerships like AWS (some even run the smaller Claude Instant models on-premises for sensitive data). The UI components Anthropic has built (Slack bot, etc.) are generally praised for being simple and effective, bringing AI assistance into collaborative spaces.

Multi-User Collaboration & Workspaces: Claude's introduction of **Team accounts and Projects** specifically targets collaborative use. In June 2024, Anthropic announced features to *"improve team collaboration and productivity"* by adding Projects for Claude Pro and Team users pymnts.com pymnts.com. As mentioned, Projects allow sharing of content among team members. If you're on a Claude Team plan (where multiple users are under one billing and organization), you can **share conversations** into a Project's activity feed pymnts.com. For instance, if one analyst had a great Q&A with Claude about a research topic, they can make that chat visible to teammates in the Project so others can learn from it or continue the thread pymnts.com. This is a collaborative knowledge-building approach – essentially, good AI interactions don't have to live and die in one person's account; they become assets the whole team can leverage. Team members can upvote or comment on shared chats in the feed, making Claude a bit of a collective brainstorming partner.

Furthermore, the ability to **incorporate internal documents into Projects** (uploading style guides, codebases, transcripts, etc. as context) means a team can curate a set of knowledge that Claude will use for everyone's queries pymnts.com pymnts.com. This is incredibly powerful in a business setting: imagine a Project for Customer Support that has all relevant policy docs

loaded – any team member asking Claude a question in that Project will get answers that draw from *the company's actual documentation*. It effectively turns Claude into a company-specific assistant. The UI to add documents is simple drag-and-drop or selection from a drive, and you can manage those files (replace/update them as policies change). Anthropic wisely clarified that *"any data or chats shared within Projects will not be used to train our models without consent."* pymnts.com – a reassurance that internal info stays internal.

Claude's Projects also support **multi-user editing** in the sense that multiple people can contribute to the same knowledge base and see each other's interactions, though they aren't typing in the same chat bubble simultaneously. It's more asynchronous collaboration, but quite effective. It's akin to a shared folder of AI chats and resources for a team. This feature set prompted some to call Claude *"an AI built for teams"*, given how early they rolled out these sharing capabilities compared to others.

In terms of real-time collaboration, Anthropic hasn't announced something like "multiple people in one live chat instance" – that remains single-user per chat. But given the Slack integration, teams can collectively chat with Claude by adding it to a channel, which is one way to have a multi-party conversation including the AI. One could ask a question in Slack to @Claude, another team member could follow up, and Claude will see the whole threaded context and respond accordingly. This is quite similar to having a group chat with an AI participating.

Accessibility & Multi-Modal Support: Historically, Claude was text-only. However, following OpenAI's and Google's lead, Anthropic introduced **voice capabilities** in late 2024. Claude's mobile apps now allow voice input and can read answers aloud in a natural voice venturebeat.com. The approach is similar to ChatGPT's – a microphone icon to start speaking, and Claude will transcribe your query (using Anthropic's or a partner's speech-to-text). You can also choose a voice for Claude to respond with. Anthropic offered a handful of voice options (e.g., a friendly female voice, a clear male voice, etc.), though not as many as Google's. These voices are described with terms like *"Buttery"* or *"Rounded"* for their tone venturebeat.com.

Claude's voice mode on mobile is notable for how it integrates with content: as mentioned, it can fetch personal data (calendar events, emails) and *read them out loud* summarizing them venturebeat.com. It also provides **visual text transcripts and summaries** of voice conversations venturebeat.com – after a voice session, the Claude app displays the full text transcript and even a bulleted summary of key points from the discussion. This is great for accessibility (deaf users can get the text, or if you missed something Claude said, you have a transcript) and for productivity (you have meeting notes automatically). The UI design here is thoughtful: transcripts appear in the chat as if they were part of the conversation, and you can continue the conversation in text from that point if desired.

Regarding images and other modalities, Claude remains primarily a text-based model with no native image understanding or generation announced as of June 2025. If you give Claude an image, it won't directly interpret it (unlike ChatGPT or Gemini). So for now, users needing image analysis would have to use a companion tool or a different platform. Anthropic might be working

on multimodal capabilities (given industry trends), but they've been cautious. On the other hand, Claude can **output certain rich content** – it can produce Markdown tables, formatted text, or JSON, and with the Artifacts feature, even simple graphics (like if asked, it can suggest the code for a flowchart in Mermaid markdown, which could be rendered externally).

For multilingual support, Claude was initially heavily English-optimized, but it has improved in other languages. It can understand and respond in languages like Spanish, French, German, etc. fairly well, though generally not as robustly as Google's model for nuanced local idioms. Anthropic hasn't marketed multilingual as a key strength, so business users requiring strong non-English support might lean towards Gemini or open-source local models tuned for their language.

Anthropic has ensured their UI is accessible in terms of design – it supports screen readers (the web interface uses proper labels so visually impaired users can navigate), and the simple interface inherently works with high contrast or zoom. The mobile apps similarly follow platform accessibility guidelines for larger text, VoiceOver/TalkBack compatibility, etc. In summary, **Claude focuses accessibility on voice and readable formats**, but lacks the image/vision component present in some competitors.

Handling Long Documents, Code, and Tabular Data: As noted, long document handling is one of Claude's standout features. With a context window of up to 100K tokens (and 200K in some modes), Claude can absorb hundreds of pages of text in one go pymnts.com. Business users leverage this by feeding large reports, entire books, or massive datasets into Claude for summarization or analysis. The Claude UI even allows **multiple file uploads at once** – you can drop, say, five PDFs into the chat or project, and Claude will ingest all of them (up to the context limit) to answer questions. This batch upload is something ChatGPT's interface, for instance, doesn't natively do (ChatGPT requires using the Code Interpreter plugin one file at a time). The ability to **bring huge context to a conversation without external tooling** is a huge time-saver. Claude effectively can be your reader for lengthy compliance documents, technical manuals, or code repositories.

Speaking of code: Claude is quite skilled at code generation and debugging. In fact, many users find Claude's coding style very clear and well-commented. The UI highlights code in monospaced blocks and, like others, has a copy button for convenience. The **Artifacts** feature in Claude's UI really shines for coding use-cases: if Claude generates a long piece of code or even multiple files, it can package them as an Artifact bundle that you can download as a zip or view in a structured way. For example, if asked to generate a simple website with HTML, CSS, and JS files, Claude might provide a link to an Artifact that contains all those files neatly, rather than dumping all code in one long message. This is a very developer-friendly approach and reduces clutter in the chat.

For **tabular data**, Claude will produce Markdown tables if appropriate. With its large context, it can also analyze big CSV data pasted into the chat (up to the limit). It's not uncommon for someone to paste a 1000-line CSV excerpt and ask Claude for insights; it can manage that

where others might struggle. The UI scrolls horizontally for wide tables and vertically for long tables, and it remains responsive. Also, in Projects, if you uploaded, say, a CSV as an Artifact, you can ask questions about it without pasting it – which is even smoother.

Claude does not run code by itself (no built-in execution environment accessible to users yet), but it often provides runnable code solutions that users can copy out. Anthropic has hinted at exploring code execution safely, but they've been conservative on enabling that in the public UI due to security considerations. Instead, they lean on integration: e.g., pairing Claude with tools like Replit's coding environment (some tie-ups exist there, where Replit's Ghostwriter can use Claude under the hood).

Security & Privacy: Anthropic positions Claude as an **enterprise-friendly AI**, and its UI and features reflect that emphasis on privacy and control. Claude Team and Claude Enterprise (for larger orgs) ensure that sensitive data stays protected. By default, Claude does not use customer-provided content from [Claude.ai](https://claude.ai) for training its models (similar to OpenAI's policy for enterprise). The Projects feature was launched with clear privacy commitments: *"data or chats shared within Projects will not be used to train...models without explicit consent."* pymnts.com This implies Anthropic might offer an opt-in if a company *wants* to allow using their data for fine-tuning a custom model, but otherwise it stays out of Anthropic's training pipeline.

Claude's UI includes a **redaction and filtering system**: If a user attempts to input extremely sensitive personal data or certain regulated info, Claude will actually warn or refuse, depending on policies. For instance, typing social security numbers triggers a caution. This is part of Anthropic's Constitutional AI approach – the interface sometimes politely stops the user (and itself) from going into problematic areas. While not a "privacy control" in the traditional sense, it does protect users from accidentally generating inappropriate content that could lead to breaches (like it won't output someone's private info it doesn't have, and it avoids slurs or libelous content).

From an **admin perspective**, Anthropic likely provides companies using Claude Team with some control panel (though details on this are less public than OpenAI's). At minimum, they provide usage analytics and the ability to remove users from the team. Since Claude can be accessed via API, some larger enterprises deploy it in a virtual private cloud for maximum security (especially via AWS Bedrock, where the model is in Amazon's controlled environment).

The Claude UI itself uses encryption (HTTPS) and sessions tied to email login (Anthropic supports Google login, etc., for user convenience). On mobile, FaceID/biometric app lock is available for the Claude app, preventing unauthorized access to your chat history if your phone is unlocked.

An interesting unique aspect: **Claude exposes some reasoning** behind responses in a feature for developers. Not exactly chain-of-thought to end users, but Anthropic has an experimental "show reasoning" for certain debugging scenarios. It's not mainstream in the UI, but indicates Anthropic's ethos of transparency. This was in contrast to DeepSeek which shows chain-of-

thought to all users; Anthropic only does it in controlled ways, likely to avoid any accidental sensitive info leakage from the model's thoughts.

Desktop vs Mobile: Claude provides a consistent experience with its web interface and mobile apps (available on iOS and Android). The **web app** at claude.ai is full-featured. The **mobile app** has the advantage of voice mode and uses a chat-centric design optimized for smaller screens – with a hideable sidebar and larger tap targets for microphone, etc. The mobile apps launched a bit later (Anthropic released them in 2024 as it expanded consumer availability). They allow offline access to recent chats when your device is offline (you obviously can't get new answers, but you can read past ones, which is handy for reference on the go). One small difference: on mobile, Claude's app allows quick voice notes – you can record a question and have it transcribed, which some users find faster than typing on a phone.

There isn't a native Claude desktop application (no .exe or .dmg) as of mid-2025, but the web covers that need well. Some users do run Claude's web UI as an Electron wrapper to dock it like an app. The Slack and other integrations also serve as "desktop" access in a way, since many keep Slack open all day.

Performance-wise, on both desktop and mobile, Claude's response streaming is very fast for the first part of an answer (Anthropic optimized for quick starts) and then steady for long outputs. If using the 100K context with a huge document, there might be a slight delay while it processes internally, but then it streams out the summary fluidly. The UI indicates when it's "*thinking*" with an ellipsis animation.

In conclusion, **Claude's interface** stands out for its support of extremely large-scale conversations and collaborative knowledge integration. It feels like a smart collaborator that can juggle whole project files and discussions at once. While perhaps not as flashy or multimodal as some competitors, for many business scenarios Claude's thoughtful UI features (Projects, long context, shareable chats, etc.) make it a powerhouse for teams needing deep analysis and a shared AI helper.

Poe (Quora) – The Multi-Model Hub

Quora's **Poe** takes a different approach from the single-model assistants by positioning itself as an **aggregator of many AI models**. Its user interface is built to let you chat with a variety of bots (like GPT-4, Claude, etc.) in one place, and even **compare or converse with multiple bots at once**. This makes Poe's UI quite unique and useful for power users who want flexibility.

Layout and Design: Poe's interface on web and mobile is centered on a chat area, but unlike others, it has a top menu or sidebar for selecting which *bot/model* you are chatting with. The design is colorful – each model or bot might have an icon (GPT-4 has OpenAI's logo, Claude has Anthropic's logo, etc., and user-created bots have custom emojis). The overall aesthetic is clean but a bit more "**busy**" than ChatGPT or Claude. As one comparison put it, Quora's Poe UI can

feel “*more cluttered and utilitarian*,” with lots of navigation elements visible [magai.co](#). On the left side (or a top bar on mobile) you have a list of available bots and your current conversations with each. Poe tends to treat each bot as its own conversation thread by default – e.g., you have one ongoing thread with GPT-4, one with Claude, etc., which you can reset at any time. There is a **new chat** button if you want to start a separate thread with the same bot (like to keep different topics separate), and those will show up as sub-conversations under that bot. The UI also shows you a feed of **community-created bots** you can try, which gives it a bit of an app-store feeling in addition to chat.

One standout UI element is Poe’s **multi-bot chat** feature. In a single conversation, you can actually add multiple AI bots and have them answer in turn. The interface for this shows multiple AI avatars within one chat and displays each model’s answer side by side or sequentially [allaboutai.com](#). For example, you can ask a question and have both GPT-4 and Claude respond, allowing you to compare their answers directly in one view. This is a researcher’s dream and also useful for validating important answers (if both models agree, you gain confidence; if they differ, you can investigate why). Poe introduced this multi-bot capability in 2025, and it’s exposed in the UI with an option like “Group Chat” where you select which models to include, then the chat messages are color-coded or labeled by model name [allaboutai.com](#). The layout might show them side-by-side on a wide screen (for direct comparison) or one after the other with clear headings on smaller screens.

Ease of Use: For basic chatting, Poe is straightforward – pick a bot and start typing. It’s designed to be **beginner-friendly**, with an “*intuitive interface*” and modern chat app features [allaboutai.com](#). There are handy conveniences: input boxes have a “**suggested prompts**” feature (when you select a bot, Poe might show examples of what you can ask it), and you can swipe between bots’ conversations easily on mobile. The slight complexity comes from the abundance of choice; a new user might be unsure whether to ask ChatGPT or Claude or Poe’s own model “Sage” a question. Poe tries to guide users by highlighting certain bots (like GPT-4) as “recommended for complex tasks” or showing trending community bots.

One nice ease-of-use feature is **cross-platform sync**. Poe is available on iOS, Android, Mac (an Electron-based desktop app), and web, and your account ties them together. Your chat history with all bots is stored in the cloud and “*persistent...cross-platform*” [linkedin.com](#), so you can start a conversation on your phone and continue it on your laptop seamlessly.

Poe also handles the quirks of each model automatically – for example, if a model like GPT-4 has a message limit (per month, etc.), Poe shows how many uses you have left or disables it until the quota resets, rather than failing mid-conversation. The interface spares the user from needing to manage API keys or anything; Quora takes care of that behind the scenes.

Prompt Management & History: Poe keeps a running **chat history** for each model you use. On the sidebar, under each model’s name, you might see your recent conversation topics. You can tap one to reopen it. Poe also allows you to **delete chat history** if desired [allaboutai.com](#), either

per conversation or all data (they provide a straightforward way to wipe your account's conversations, which some privacy-conscious users appreciate).

However, Poe does not yet have advanced organization features like folders or project groupings – the assumption is each conversation is relatively short or one-off. It's more akin to a messaging app where you have different contacts (the bots) and each contact has a chat thread. For prompt management, Poe's differentiation is more on customizing bots rather than managing long threads.

That said, one aspect of "history" in Poe is the **model context memory** depends on the model itself – e.g., GPT-4 on Poe will remember the last ~8K tokens in that thread. If you exceed that, older messages roll off context, but they remain visible in the UI for your reference. There's no warning when context is full; it's up to the user to reset if needed. Many Poe users will reset the conversation with a bot for a fresh context when starting a very different topic (there's a clear reset button for each bot).

Poe also doesn't (as of June 2025) have a search-in-history feature. Given the persistent logs, power users might like that, but it's not implemented yet. Possibly because many use cases on Poe are ephemeral Q&As rather than building a long knowledge base.

Customization Options: One of Poe's biggest attractions is the ability to **create your own bots** with custom instructions or behavior. Through Poe's UI, users can go to the **"Create a Bot"** section, where they can choose a base model (GPT-4, Claude, etc.) and then enter a custom prompt that serves as the bot's persona or instructions. They can also give it a name and an icon. This no-code bot creation is extremely easy – just describe what you want (e.g., *"This is an AI that speaks like Shakespeare and only talks about economics"*) and Poe will generate a bot for you allaboutai.com. These bots can then be kept private for your own use or **shared publicly** with the Poe community.

In fact, Poe has a **community bot marketplace** of sorts. Users can browse and discover **"Community Bots"** created by others allaboutai.com. For example, you might find a bot for "Legal Advisor" or "German Translator" created by another user and try it out instead of crafting your own prompts from scratch. This harnesses crowd-sourced prompt engineering and makes it readily accessible via the UI. Each community bot has a page with a description and usage stats, and you can follow updates if the creator tweaks it.

For business users, this feature means internal experts could create specialized bots (like a bot fine-tuned for their product info) and share it with the team (though note: sharing on Poe is currently global, not limited to an organization unless you keep it private). If a company wanted to, they might have employees use a custom Poe bot for consistency in answers. Poe even introduced a **monetization program** where bot creators can get paid based on usage allaboutai.com. While more relevant to hobbyist creators, it indicates the platform's incentive to encourage high-quality custom bots – which benefits users by having a rich library to choose from.

Aside from custom bots, Poe doesn't have much in terms of toggling the model's behavior (since each underlying model has fixed parameters). But you do get to pick from **over 100 models** including some image generators and others allaboutai.com. So customization is often just switching to the model that fits your task. The UI for model selection is straightforward, either via a list or a carousel of model icons.

Integration with Business Tools & APIs: Poe is more of a self-contained app and doesn't natively integrate with third-party business tools like Slack or Google Docs. Its focus is on providing the models themselves. However, **Poe did release an API** that allows developers to integrate Poe's bots into their own applications linkedin.com. The Poe API essentially lets you send a query to a specific bot (including user-made bots) and get the response, leveraging Poe's infrastructure. One reason a business might use the Poe API is to have a simpler way to access GPT-4 or Claude without managing multiple API keys or infrastructure – Quora handles the scaling. Plus, if you've fine-tuned a perfect prompt as a Poe bot, the API lets you deploy that easily. Poe highlights that the API comes with benefits like *"distribution to millions of users already using Poe"* and *"persistent user history"* if you build a bot through Poe's system quorablog.quora.com. So a company could theoretically build a Poe bot and share it with a large audience as a kind of Q&A or customer service tool (though for truly sensitive or bespoke integration, most might use direct model APIs).

Poe also integrates multiple modalities: it includes some **image generation models** (like Stable Diffusion) accessible via the same chat interface allaboutai.com. So, within Poe, you could switch to an image model, type a prompt like "logo idea with X and Y", and it will return an image. These images show up in the chat as images you can enlarge and save. Poe's addition of multimodal models (even some that generate audio or video) means it's a bit of a one-stop-shop for various AI tasks. For instance, DALL-E 3 was integrated, so business users could quickly create an illustration by just changing the bot from GPT-4 to DALL-E in the UI. There's no direct integration into say PowerPoint, but you can easily copy results over.

Multi-User Collaboration & Workspaces: Poe does not have collaborative features in the sense of multiple people sharing a chat or a team workspace. It's oriented towards individual use. There isn't a concept of a Poe for Teams with shared chats or admin controls. If collaboration is needed, users usually have to copy outputs or share screenshots.

One quasi-collaborative aspect is the **public profiles and Q&A**. Since Poe is made by Quora, they have an experimental feature where people can ask questions on Quora and get an AI draft answer (with sources) via Poe, which can then be edited by humans before posting on Quora. But that's more community Q&A than direct teamwork.

However, the community bot sharing is a form of collaboration – prompt creators and users interacting. You can think of it as an open marketplace where one user's "prompt engineering" benefits another. Bot creators often iterate based on feedback, effectively collaborating with their user base to improve the bot.

For a business wanting to use Poe internally, the lack of private multi-user spaces is a limitation. Poe is more of a personal productivity and exploration tool rather than an enterprise solution. That said, an organization could instruct its users to use a certain Poe bot for guidance or knowledge, but that's informal.

Accessibility: Poe's apps and website are polished (Quora has years of experience with web communities, and it shows). On iOS, for example, Poe supports VoiceOver and dynamic text sizing, making it accessible to visually impaired users. The design uses sufficient color contrast for most text. Poe also implemented **multilingual UI** to some degree, but since it's largely user-driven content, the interface itself remains English-centric with support for input/output in many languages via the models. Poe explicitly added **enhanced multilingual support** in 2025, meaning the app can properly handle input in languages like Japanese, French, etc., and the display fonts and text directions (for RTL scripts) are correctly rendered allaboutai.com.

As for voice, as of mid-2025, Poe doesn't have built-in voice input or output. It's something users have requested, but currently you have to use your keyboard to interact. If someone uses Poe on mobile and wants voice, they might rely on the phone's keyboard dictation, which works but isn't the same as integrated conversation mode. So on that front, it lags behind ChatGPT and Claude.

Support for Long Documents, Code, and Data: Poe's ability to handle long inputs depends on the chosen model. Poe itself doesn't implement special tools for chunking or uploading files (there's no file upload UI in Poe). If you want to feed a large document, you'd paste it in parts or rely on a model that can browse a link (for example, Poe's GPT-4 cannot browse, but Poe offers another bot called "WebGPT" or similar for retrieval tasks). This is not as seamless as giving ChatGPT a PDF or Claude a long doc directly. For coding tasks, Poe again relies on underlying models like GPT-4 or Claude which are good at code. The UI will display code with formatting, but unlike ChatGPT, there's no execution sandbox in Poe's interface. It's purely for generating or debugging code via text.

However, one cool feature: Poe's **multi-bot threads** can be leveraged for long content – you could have one bot summarize part of a document and another verify or expand on it within the same thread, dividing the labor. This is an unconventional but effective way to use multiple AIs on a single complex input.

Security & Privacy Controls: Quora's Poe is a consumer-facing product, and while it values user privacy, it doesn't have the explicit enterprise-grade guarantees of not training on data. In fact, Quora likely uses interactions to improve their own models (like their "Sage" bot). Poe's privacy policy is transparent that data may be stored and reviewed to monitor the service. Users who need to avoid that would use the delete history feature often allaboutai.com. Poe does secure all connections and uses authentication (Quora account or Apple/Google login) to sync your data.

One thing to note: when using Poe, queries to models like GPT-4 or Claude are being relayed via Quora's servers to OpenAI/Anthropic. So you have to trust Quora as an intermediary with your content. For many casual users that's fine, but businesses might hesitate to send proprietary info through a third-party aggregator. Poe currently doesn't offer an on-prem or private instance.

There is a **blocking and filtering** component in Poe: they enforce OpenAI's and other providers' content rules, so if you try to get a disallowed output, you'll get a message that the bot can't comply. This is standard and ensures that publicly shared bots can't be easily abused to produce extremely harmful content under Quora's name.

All in all, **Poe's interface** excels in giving users choice and community-driven customization. It is great for individual power users and small-scale use, especially if you want to pit different AI models against the same problem or quickly spin up a specialized assistant. In a business context, Poe can be a playground to experiment with various AI capabilities and even host a custom bot for customers (some people embed Poe bots on their sites via the API), but it's not a dedicated enterprise solution with collaboration or guaranteed privacy. It's more analogous to having a Swiss army knife of AI models in one app, with the convenience and complexity that entails.

Llama (Open-Source Solutions) – DIY and Highly Customizable

"**Llama**" in this context refers to the open-source large language models developed by Meta (LLaMA, LLaMA 2, etc.) and the myriad user interfaces built around them by the community. Unlike the other names on this list, Llama doesn't point to a single product or app – instead, it signifies the **ecosystem of open-source LLMs** and the UIs that allow business users to interact with those models. Many businesses opt for open-source LLMs for data privacy, cost, or customization reasons, and there are several interface options available (from Hugging Face's **HuggingChat** to self-hosted web UIs like **Chatbot UI** or **NextChat** belsterns.com). We'll discuss the general characteristics of these open-source UIs relevant to Llama-based models.

Layout and Design: Most open-source chat interfaces aim to emulate the clean, simple design of ChatGPT – a large chat transcript area and an input box. **HuggingChat**, for example, has an interface that looks very much like ChatGPT: conversation bubbles for user and assistant, and a left sidebar listing past conversations. It's intentionally familiar so that users can jump in without a learning curve. Another popular interface, **Chatbot UI** (an open-source project on GitHub), offers a ChatGPT-like customizable front-end that companies can self-host belsterns.com. It likewise features a straightforward layout with the ability to support multiple backend models. Generally, open-source UIs might have a few more visible toggles (because they cater to power users). For instance, you might see options to adjust the **model parameters** (like temperature,

max tokens) in a settings panel – things proprietary UIs keep hidden. The design can often be tweaked as well; since code is open, organizations can brand the interface with their logo or colors if desired belsterns.com.

One thing to note is that because there are many UIs, the consistency varies. Some projects like **NextChat** focus on minimalism and responsive design (so it's easy to deploy and works on mobile out of the box) belsterns.com. Others like **Hugging Face Chat** highlight the model selection aspect, with lists or galleries of models you can try belsterns.com. This "model explorer" approach means the UI might show a library of models to choose from before you start chatting belsterns.com – which can be very useful in finding a model that suits your task (be it a code-focused model, a small fast model, a multilingual model, etc.).

Ease of Use: For end-users (non-developers), open-source chat UIs have become quite user-friendly, though getting to the point of using one might require some initial setup if self-hosting. If using a hosted solution like HuggingChat on the web, it's as easy as any web app – go to the site and chat. HuggingChat even has an **official app for iOS** now apps.apple.com, making it accessible like a typical app store offering. The advantage of something like HuggingChat is that it comes preloaded with several **"assistant"** style open models (e.g., Llama-2-Chat variants, OpenAssistant, etc.), so users don't have to know how to configure anything.

For a business user who isn't technical, a self-hosted UI might be initially daunting (setting up a server or installing an app). But projects like Chatbot UI have **one-click deploys (on Vercel, Docker, etc.)** belsterns.com that IT can handle, after which using the chat is trivial for the employees. Many of these UIs allow login or some access control, but often they're kept simple without requiring accounts unless you add one.

One feature aiding ease-of-use is **multi-model compatibility**: UIs like Chatbot UI or others can connect to different providers – OpenAI API, local Llama models, Anthropic API, etc. belsterns.com. This means if a user is comfortable in that UI, they can use it to talk to various models similarly. For example, a developer might integrate both a local Llama-2 13B model and a remote GPT-4, and the UI will have a dropdown to switch between them. This flexibility is powerful, though it might confuse non-technical users if not presented cleanly. Ideally, an admin would configure a default model, so regular users don't have to think about it.

Prompt Management & History: Most open UIs offer basic conversation history (since that's expected now). HuggingChat introduced multi-conversation support early on. If you log in with a Hugging Face account, your chats are saved in the cloud and you can revisit them. The UI will list them similarly to ChatGPT's sidebar. Self-hosted solutions typically store history in local storage or a small database, enabling persistent conversations. For example, Chatbot UI stores chats in your browser by default, or can be configured to use a database for multiple users, and it supports renaming chats and deleting them.

One downside often cited is that open models have **smaller context windows** by default (Llama-2, for instance, typically has 4K tokens). This means the model might forget earlier parts

of a long conversation more quickly than Claude or GPT-4 would. Some open UIs mitigate this by implementing **recency-based truncation or summarization** – e.g., automatically summarize earlier messages to free up space. These are not universally present, but some advanced community UIs allow plugin-like extensions to handle longer context via retrieval.

Open-source interfaces also allow more **prompt control** to the user. Many will let you edit the system prompt (the hidden initial instruction to the model) if you want. This is an expert feature but can be useful to set a certain style or policy for the model globally (like instructing it “Always answer in corporate style”). In proprietary UIs, system prompts are fixed or abstracted via settings; in open UIs, you often have direct access if you enable it.

Customization: This is where open-source shines. Because the code is accessible, companies and individuals can **tailor the UI and the model behavior extensively** belsterns.com. Need a company logo and custom color scheme? Easily done by tweaking the HTML/CSS. Want the AI to integrate with a company database? A developer can modify the backend to call an API or use a vector store for retrieval augmentation. Some open frameworks have plugin architectures – for instance, there are community add-ons for things like **web search and PDF parsing** integrated into UIs like HuggingChat github.com. HuggingChat’s GitHub indicates it has a “*powerful Web Search feature*” that generates search queries and fetches results if you enable it github.com, similar to how Bing works. It also has or plans tools for documents. So with open UIs, if a business needs a particular integration, they can either find an existing open plugin or build it.

Fine-tuning the model is another level of customization relevant to Llama. While not done through the chat UI per se, a company might fine-tune Llama-2 on their data, then deploy it behind the UI. The interface is model-agnostic, so it will happily converse using the fine-tuned weights. This is something closed platforms don’t allow (you can’t fine-tune GPT-4 yourself, but you can with Llama). The UI might expose a way to switch between the base model and fine-tuned model if both are loaded.

For **user-specific customization**, open UIs might not have slick features like ChatGPT’s custom instructions panel, but you can approximate that by editing system prompts or running a private instance per user with their preferences baked in.

Integration with Business Tools: By default, most open-source UIs are standalone and don’t come pre-integrated with third-party business apps. However, because they are open, you can integrate them if needed. For example, a company could embed an open-source chat UI into an internal portal or intranet. They could connect it to Slack via a bot that calls the local model (some have done this using libraries like llama.cpp or text-generation-webui API). The key difference is, these integrations are **DIY** – there isn’t an official Slack plugin from “Llama, Inc.” because it’s community-driven.

One notable tool is **LlamaIndex (GPT Index)** which many have used to connect LLMs with external data. While not a UI itself, LlamaIndex can provide a back-end for retrieval, and the

front-end could be one of these open chat UIs. There are examples where developers built a simple company Q&A chatbot by combining Llama-2 with a document index and a basic web UI.

For businesses, a big integration advantage of open-source is **API flexibility without usage caps or fees** – if they self-host the model, they can integrate it into as many workflows as they want, with costs limited to their infrastructure. They aren't constrained by per-query pricing or data egress limits of an external API. This is why some have called open solutions *"ideal for companies needing specialized model access and control"* belsterns.com.

Multi-User Collaboration & Workspaces: Out-of-the-box, open UIs like Chatbot UI or NextChat don't include multi-user account management (they're often one-user applications). But because you can put them on a server, multiple people can technically connect to the same interface. If not designed for that, it might mean everyone sees the same chat history, which isn't ideal. Some projects have added multi-user support; for example, an enterprise might fork an open UI to add login and separate chat spaces per user. Or they may deploy one instance per user via containerization. It's more effort than something like ChatGPT Teams where it's built-in, but it's feasible.

For explicit collaboration features, open UIs generally don't have things like shared chats or feeds. However, if a company wanted a collaborative aspect, they could implement something custom (like a shared "wiki chat" where all can see and contribute). The open nature gives freedom, at the cost of requiring developer input.

Accessibility: Open-source UIs benefit from community contributions in accessibility. Projects on GitHub often accept improvements for ARIA labels, keyboard navigation, etc. HuggingChat likely adheres to basic web accessibility standards (Hugging Face tends to be mindful of that). Also, some specialized open projects focus on voice interaction – e.g., **SpeechGPT** which is an open framework combining speech recognition and TTS so you can have voice conversations with local models belsterns.com. That project specifically highlights features like multiple STT/TTS backend options and full self-hosting for privacy belsterns.com. A business could deploy a voice-enabled open chat UI for accessibility or interactive kiosk purposes, without relying on a cloud service.

Language support is strong in open models that are trained multilingually or region-specific models (like XLM variants or Mistral, etc.). If a business needs an AI that speaks a less common language, open-source might be the only option or at least significantly better. And the UI itself can often be translated if it's open; one could localize the interface text to Spanish or Chinese by editing a config, something closed UIs may not offer.

Handling Long Documents, Code, Tabular Data: Out of the box, Llama-based models have limitations (e.g., 4k context). But the community has developed strategies like **retrieval augmentation** to allow handling of long documents. Many open UIs support plugins or have extensions for uploading a PDF or text file and then the UI will split it and feed relevant chunks to the model as you ask questions. For instance, there's a mention in open communities of

HuggingChat adding PDF support [reddit.com](https://huggingface.co/docs/chat-ui/en/adding-pdf-support). So, while the base model can't ingest a 100-page PDF at once, the UI+tooling can make it appear as if it does by on-demand searching the document.

For coding, open UIs can integrate with tools like a local Jupyter kernel to execute code. Some advanced ones (especially if using a smaller model that runs on CPU) might even allow code execution inline. But typically, the UI just gives you the code answer and you have to run it elsewhere. However, since you can modify these UIs, a company could wire up a sandbox and connect it – it just requires development work.

One well-known open approach is using **VS Code extensions** with Llama models. For instance, there's a CodeLlama VS Code plugin that provides an in-editor chat using Llama-2. That's a form of UI too – specifically integration into a developer's IDE, which many prefer for coding tasks over a separate chat app. It highlights that open LLMs can be embedded in various user interfaces beyond a web chat: IDEs, command-line assistants, mobile apps, etc., thanks to their open licenses and smaller size.

Security & Privacy: For businesses, the biggest appeal of using Llama (open source) is **complete data control**. If you run the model on-premises or on your private cloud, none of the conversation data leaves your environment. Open UIs can be set up behind your firewall. This addresses concerns about sending proprietary data to third-party servers. As one source notes, *"DeepSeek (an open model) can run entirely on your device, ensuring your data stays private"* larksuite.com – the same is true for Llama models.

There are also no usage restrictions – you are not subject to an external provider's content policies or rate limits (beyond what you enforce). You can fine-tune the model to avoid certain information leakage and the UI can be configured to mask sensitive info if needed.

That said, open models typically don't have the refined guardrails of something like ChatGPT out-of-the-box. The UI might not warn you about biases or inappropriate requests. Businesses often have to implement their own filters or rely on fine-tuned safer variants. Some open UIs incorporate basic moderation (for example, using an open content filtering model to intercept extremely bad prompts). In any case, the security responsibility shifts more to the user/organization – they must ensure their deployment is secure (e.g., restrict who can access the UI, keep the server software updated, etc.).

One interesting development: There are open UIs focusing on encryption and secure multi-tenant use. For instance, a project like **LibreChat** markets itself as an open alternative focused on privacy where even the conversation storage can be encrypted.

In summary, using **Llama through open-source interfaces** provides unmatched flexibility and data ownership at the cost of more hands-on management. The UI experience can be nearly as polished as commercial chats (especially with projects backed by communities or companies, like HuggingChat which is backed by Hugging Face, ensuring a good user experience). Companies that want to deeply integrate AI into their systems or products often go this route

because they can tailor both the model and the interface to exactly what they need, free from external limitations [belsterns.com](#) [belsterns.com](#). For a business user, if IT sets it up well, interacting with an open-source Llama-based assistant can feel as easy as any other chat – except that behind the scenes, it's running on your own infrastructure or customized for your domain.

LeChat (Mistral AI) – The New European Entrant

LeChat is a relatively new AI assistant interface from Mistral AI (a European startup). It pairs Mistral's open large language models with a user-friendly chat UI. Launched in late 2024, LeChat is Mistral's bid to rival ChatGPT by offering a free, privacy-conscious assistant powered by their own models. As of June 2025, LeChat is still in beta, and while promising, it has some rough edges to consider.

Layout and Design: LeChat's interface at first glance is **straightforward and minimalist**. Upon sign-in, you're greeted with a simple chat screen – a blank conversation area and an input box at the bottom [superannotate.com](#). There's a top menu that allows switching between different Mistral model versions (currently Mistral Small, Mistral Large, and an experimental "Next" model) [superannotate.com](#). This is a unique feature: the UI explicitly lets you choose the AI model variant based on your needs (speed vs depth) – for example, "*Mistral Large for depth in reasoning, Mistral Small for fast answers.*" [superannotate.com](#) It effectively gives a built-in trade-off control which business users might appreciate when they want either a quick answer or a more thorough one.

The overall design is utilitarian. It doesn't have a lot of visual frills; in fact, early users commented that the interface could be more polished and **wasn't as easy to use as ChatGPT's cleaner setup** [superannotate.com](#). The color scheme is simple (light mode with white/gray background, dark text; a dark mode exists too). One notable missing piece initially was the ability to **rename or organize chats** – by default, it just kept a running list of past conversations with no renaming, which users found limiting [superannotate.com](#). Mistral's team has taken that feedback on board, so we might see improvements like chat titles or folders soon. At the moment, the UI lists conversations by the first line or timestamp, which can be confusing if you have many.

Despite these early shortcomings, LeChat's design does reflect a focus on **document analysis**. Mistral AI touted "**advanced document understanding capabilities**" in LeChat [mistral.ai](#). In practice, the interface allows you to **upload documents** (text, PDFs) for LeChat to analyze. The chat input area has an attachment button where you can add files. Once uploaded, LeChat can answer questions about the content or summarize it. This is a crucial feature for business use-cases like reviewing reports or contracts, and it's offered free. Combined with LeChat's large context window of 32k tokens [superannotate.com](#), the UI can accommodate quite lengthy documents in a single go. Early user feedback doesn't mention a fancy UI around file handling –

likely it's a basic list of uploaded files or references within the conversation – but it gets the job done.

LeChat's interface also includes an **optional web search toggle** (called "Web Search" or similar) and an **image generation tool** as per some descriptions digitrendz.blog. It's not clear if those were fully live at launch or just planned; the official word said "*Le Chat operates without internet access*" currently superannotate.com, indicating that as of early 2025, real-time search was not enabled. It might be an upcoming feature. Similarly, image generation (possibly hooking to Stable Diffusion) was hinted at in their materials. The UI screenshot referenced in a blog shows tabs for Canvas, Web, Image generation digitrendz.blog, suggesting they are experimenting with a multi-functional interface – perhaps a mode to generate images or a mode to browse web content. If those modes become active, LeChat could become a multi-modal assistant like the big players, all within one interface.

Ease of Use: LeChat requires an initial signup (Mistral accounts) and there were waitlists during the early beta superannotate.com. Once in, using it is similar to any chat AI – type and hit enter. The early **performance issues** – such as crashes or slow responses – were noted by users due to server overloads at launch superannotate.com. Over time, as the service stabilized and scaled, these have likely improved. Assuming a stable session, the interaction flow is typical: you ask a question, LeChat responds streaming.

One area LeChat needed improvement was the overall **usability and feature refinement**. Users explicitly mentioned that it "*isn't as easy to use as it could be*" and gave suggestions superannotate.com. For instance, being able to **rename chats** was mentioned – an indicator that for serious work, organization matters, and LeChat lacked it out of the gate superannotate.com. Also, it might be missing conveniences like copying code blocks or customizing the assistant's tone. Since Mistral is actively developing it, we can expect iterative improvements.

That said, LeChat is *free*, which lowers the barrier for business users to try it out. The UI doesn't hit you with paywalls, and there are no usage limits announced (though practically, heavy usage might queue or throttle, we don't have details). So ease-of-use extends to ease-of-access in this case – just sign up with an email and you have a capable GPT-3.5 class chatbot at your disposal, no credit card required.

One thing to highlight: LeChat has **no built-in internet connectivity by design** at the moment superannotate.com. While this might seem like a limitation (and it is if you need current info), it actually simplifies usage in terms of privacy and consistency – you won't get unexpected web data or cite links. For some business queries that are internal or hypothetical, that's fine. If you need web data, you currently have to bring it yourself (or copy-paste).

Prompt Management & History: LeChat keeps a history of your conversations in the sidebar, but with minimal management features. All past chats are there, and you can click to revisit them. The models have a knowledge cutoff of 2021 superannotate.com (since they're currently offline models), so the context is static unless you feed new info via prompts or file upload.

There isn't a feature like custom instructions or long-term memory yet – each conversation is separate.

As mentioned, no renaming or folder grouping was initially available superannotate.com. So prompt management is basic: if you want to reuse a prompt or scenario, you either scroll to that conversation and continue, or copy-paste from it into a new one. Given the user feedback, we may see "Rename chat" soon to at least label what each conversation is about.

LeChat does not have multi-step prompt templates or an official library of system presets (like "creative mode" vs "precise mode"). But since you can talk to it directly, users can always instruct it to behave in certain ways. It's just not automated.

Customization Options: The main customization in LeChat is choosing the **model variant** for the task superannotate.com. This is a manual but straightforward knob to turn. Mistral Large vs Mistral Next vs Mistral Small have different strengths, and letting the user pick is a form of customizing performance vs quality on the fly. Few other UIs expose this so plainly (OpenAI hides model complexity behind GPT-4 vs 3.5, but doesn't offer multiple flavors of GPT-4, for example).

Beyond that, LeChat's UI doesn't yet offer user-driven customization like persona presets or custom bots (unlike Poe's custom bots or ChatGPT's custom instructions). However, under the hood, because it's based on an open model, one can imagine Mistral might allow community fine-tuned bots or something in the future.

One form of customization relevant for enterprise is that Mistral's models are or will be **open-source** (the smaller ones already are). While LeChat itself is a hosted service, businesses could eventually deploy Mistral models internally and perhaps use a similar UI. LeChat might serve as a reference UI for Mistral's platform (in fact the site mentions "**Build on La Plateforme**" and "LeChat – Mistral" as offerings mistral.ai, indicating they have both an API platform and the chat interface). So a company wanting more customization might use the API to integrate Mistral's model into their own app rather than rely on LeChat's UI.

Integration with Business Tools & APIs: As of now, LeChat is a standalone web app (and possibly mobile, though no dedicated mobile app yet; one accesses it via browser). There are no announced integrations like plugins or API for third-party tools in LeChat itself. However, Mistral AI definitely has an API (the website references deploying on Azure, using LangChain, etc. superannotate.com). So integration is possible at the model level, not through the UI.

The LeChat UI itself does not connect to external services (recall it's offline-first: no browsing, etc., currently). That means it also keeps your data local to that session – good for privacy, but you can't, for example, have it pull your calendar or send an email.

If Mistral's roadmaps hold, we might see "Extensions" similar to Bard's that integrate with things like web search or other APIs. The snippet from iArtificial blog suggests features like *web search* and *image generation* are in the pipeline digitrendz.blog. Possibly they will treat those as toggles

or separate modes (like an “Agent” mode that can use tools). For now, business users should view LeChat as a self-contained AI assistant that you feed information to, rather than one that will fetch information for you.

Multi-User Collaboration & Workspaces: LeChat in its current form is single-user oriented. There’s no multi-user or team concept in the interface. Each account is individual.

Mistral is likely targeting enterprise clients by promoting their platform (for deploying their models) rather than providing a multi-user SaaS like ChatGPT Teams. They did brand LeChat as “**Le Chat Enterprise AI assistant**” on their site mistral.ai, implying it’s intended to be a tool businesses use. But as of now, that probably means an enterprise user can use LeChat individually to assist in work, not that LeChat offers team accounts or admin controls.

We haven’t seen features like shared chats or projects. If a team wanted to collaborate via LeChat, they’d have to share outputs manually.

Accessibility: LeChat’s interface is relatively spartan which usually bodes well for accessibility – less clutter, easier screen reader navigation. It likely supports multiple languages input/output since Mistral models are multilingual (the Large model covers French, German, Spanish, Italian at least superannotate.com and presumably English well). The UI labels might be English-centric for now (the site is in English and French given Mistral is based in France; indeed the name “Le Chat” is French, and likely the interface can operate in French fluidly).

No voice or audio features are implemented. Mistral being a new startup might not have their own voice tech integrated yet. It’s keyboard and reading.

One could argue that by focusing on text and document understanding, they cater to users who need reading assistance – e.g., someone can upload a dense PDF and have LeChat summarize it in plain language, which is a form of cognitive accessibility (simplifying complex text). And since it’s free, individuals who couldn’t pay for GPT-4 to do this now have an alternative.

Handling Long Documents, Code, and Data: Mistral’s LeChat boasts a **32k token context window** superannotate.com, matching the upper range of GPT-4’s context (and exceeding Claude 2’s 100k in terms of raw count, though 32k is roughly 50 pages of text). This means LeChat can accept long user inputs or file content without losing context. It’s explicitly pitched for document analysis – so yes, you can paste or upload a long document and ask for summaries, Q&A, etc., all in one go without chunking it manually. This is very useful for business documents (contracts, whitepapers, logs, etc.).

The accuracy of summarization will depend on Mistral’s model capabilities, which are improving but might still be slightly behind GPT-4. Nonetheless, having a large context for free is a big draw.

For code, Mistral Large has demonstrated good coding proficiency superannotate.com. LeChat doesn’t have an execution environment (similar to others, it will only *write* code, not run it). But it can handle code-related queries. It likely formats code in monospace, though it might not have a

fancy copy button or syntax highlighting initially (user feedback didn't mention those details). If LeChat lacks these minor UI niceties, they're probably on the to-do list.

Interestingly, Mistral models were shown to outperform Llama-2 70B in some coding benchmarks superannotate.com, so LeChat could be quite a handy coding assistant. Without internet, it won't directly fetch libraries or documentation, but for language/syntax and basic algorithms it's capable.

For tabular data, LeChat can output text tables or CSV-style responses if asked. It doesn't have an integrated spreadsheet view or charting ability. However, you could upload a CSV as a text file and ask it questions about it. Mistral hasn't highlighted any specific tabular or math tool, so presumably it's all pure LLM reasoning (which can handle a bit of data, but might struggle with very large tables without a tool).

Security & Privacy: As a European company, Mistral AI emphasizes privacy. LeChat running without internet access means it won't unexpectedly call external APIs or send your data out (beyond to Mistral's servers themselves). All processing is on their server instance of the model, which they claim does not have training data beyond 2021, so it won't incorporate or leak newer info. Mistral has also open-sourced smaller models (7B) and there's discussion on openness of the larger ones superannotate.com, which suggests an orientation towards transparency.

However, one critique is that LeChat is *hosted on Azure servers* currently (according to some community notes) reddit.com. That means user queries are going to the Azure cloud in presumably EU datacenters. For some European businesses, that might raise questions about GDPR and relying on an American cloud provider, but Azure EU is generally GDPR-compliant and Mistral likely ensures data is kept secure. Still, a truly privacy-paranoid user might want an option to self-host the model. That's not what LeChat provides (LeChat is the hosted service; for self-hosting, one would get the model weights separately).

Since LeChat is free, one should consider that Mistral might monitor usage or collect feedback to improve their models (similar to how OpenAI used ChatGPT usage early on). They have a **Privacy Policy** linked on the site deepseek.com – worth reviewing for businesses. They likely anonymize and aggregate, but it's something to note.

On the user-facing side, LeChat does not have explicit admin controls or encryption options for user content. It's as secure as using any web app with SSL. If confidentiality is paramount, one might wait for Mistral to release an on-prem version. But for general use, it's reasonably secure (especially given EU's stricter regulations guiding a company like Mistral).

Desktop vs Mobile Experience: LeChat is accessed through the web (desktop or mobile browser). It is mobile-responsive, but the experience might not be fully optimized for small screens. There isn't a dedicated LeChat app in app stores yet. On a phone browser, you can log in and chat, but some UI elements (like viewing that list of models to pick or uploading files) might be a bit fiddlier on mobile. The Mistral team might release mobile apps in the future, but at this beta stage they haven't announced any.

On desktop, LeChat works in modern browsers and the interface, while basic, is sufficient. Because it's less feature-rich, it's actually pretty fast and lightweight on the web.

To sum up, **LeChat** is an evolving interface that already covers many fundamentals (free access, large context, file analysis, multiple model options) but is still catching up in polish and advanced UI features. It holds particular appeal for European businesses or any users looking for a *no-cost, privacy-aware alternative* to ChatGPT for standard tasks. The trade-off is dealing with a beta product: occasional quirks, a slightly barebones UI, and rapid changes as Mistral improves it. The company's quick responses to feedback (for example, acknowledging UI improvement needs superannotate.com) are a good sign that LeChat will only get better and more user-friendly in the coming months.

DeepSeek – The Transparent Reasoner

DeepSeek is an AI assistant that emerged from a Chinese AI startup and quickly gained international attention for its unique approach to the user experience. It offers both a powerful model (notably **DeepSeek R1**, known for strong reasoning) and a suite of interfaces: web, desktop, and mobile apps larksuite.com. DeepSeek's UI is particularly notable for its **transparency** – it shows users the intermediate "chain-of-thought" reasoning the AI goes through, which is quite unlike other chatbots. Let's break down DeepSeek's interface and features:

Layout and Design: The DeepSeek web interface follows a familiar chat layout with some extra bells and whistles. The main chat window is flanked by a sidebar that contains options like **"New chat"** and toggles for special modes (DeepSeek has **DeepThink** and **Search** toggles) larksuite.com. The design uses a dark theme by default with neon blue accents (at least in some versions), giving it a bit of a hacker aesthetic which some users find cool and others might find less corporate. However, overall it's user-friendly – you have a clear text input area, the model's responses appear in bubbles, and there are buttons or switches to activate additional functionality.

The standout UI element is when DeepSeek responds, it often first outlines *what it understands you want and how it plans to solve it* before giving the final answer platformer.news. This is the chain-of-thought display. Visually, this might appear as a separated block of italic text or a different color text that says something like: *"I think the user wants to do X. Steps: 1) do this, 2) then do that."* Then it proceeds with the answer. Users have described it like the assistant "telling you their plan before executing it" platformer.news. It's as if you see the AI's scratch paper. This UI design was bold – Casey Newton from Platformer noted that **DeepSeek's decision to expose its chain of thought was a "surprise hit"** and could influence other AI products to follow suit platformer.news platformer.news. For business users, this feature can instill more trust (you see *why* it's giving a certain answer) and also allows them to correct the AI mid-way if the plan looks off.

Ease of Use: Getting started with DeepSeek is straightforward and *free*. The company provides a web chat (no lengthy signup required for the basic usage – currently, it’s open access). They also offer **desktop and mobile apps** which mirror the web experience larksuite.com. The presence of apps means you can use DeepSeek offline (for the local model) or on the go easily. Installing the desktop app (likely an Electron app) or mobile app from app stores would give a more native feel, like any messaging app.

Within the interface, users have control features like **“New Chat”** (to start a fresh conversation and context) and the aforementioned toggles:

- **DeepThink mode:** When toggled on, DeepSeek will take extra steps to reason more deeply or systematically about your query larksuite.com. In practice, this results in a slower but potentially more thorough response, and you might see a longer chain-of-thought presented.
- **Search mode:** When toggled, DeepSeek will actively perform web searches to find information to answer your question larksuite.com. The UI likely shows this by listing the search query it used and then providing results or citations in the answer.

These toggles make the interface a bit more complex than a simplistic chat (the user has to know when to use them), but they are clearly labeled and come with tooltips. The design is such that novice users can ignore them (DeepSeek will still answer without them), but power users can leverage them for better results on hard queries.

DeepSeek supports **multi-turn conversations**, with memory of past context similar to others (the exact context length depends on the model variant, but R1 might be around 8k tokens, V3 possibly more as it evolves). The app includes a sidebar listing older conversations larksuite.com and the ability to find them easily.

One key ease-of-use advantage: **Offline functionality**. DeepSeek can run entirely on your device (with their downloadable models) larksuite.com. They advertise that no constant internet connection is needed and data stays private offline larksuite.com. For a business user, this means you could use DeepSeek on a local machine for sensitive work without network concerns (though you’d need a capable machine to host it, or use the mobile/desktop app in offline mode with smaller models). The UI likely has an indicator if it’s in offline mode vs cloud mode. The default web usage is cloud (their servers), but the app might allow switching to a local model if downloaded.

Prompt Management & History: The DeepSeek interface includes a conversation history sidebar where all your past chats are accessible larksuite.com. Each chat is timestamped and perhaps partially titled by the first user query. You can click to reopen them, and continue where you left off. This is quite standard.

What’s interesting is how the chain-of-thought might or might not persist across turns. Typically, chain-of-thought is internal per turn, but DeepSeek shows it for each answer. So if you scroll up in history, you see not only what you and DeepSeek said, but also *how DeepSeek reasoned at each step*. It essentially doubles as a log of reasoning. This can make reviewing a conversation

later very insightful – you can follow the logic that was used for each reply. It’s almost like having a built-in “explain my answer” after every answer.

DeepSeek doesn’t yet have a feature to label or categorize chats beyond chronologically, but you can delete conversations if needed (especially for privacy or decluttering). They emphasize privacy, so presumably deleting truly wipes it from their servers (if you were using the cloud service).

Customization Options: DeepSeek’s UI and ecosystem give users a high degree of control:

- **Model selection:** As of early 2025, DeepSeek had at least two main models: V3 (a general model) and R1 (a reasoning-optimized model). The UI likely allows choosing which to use. The homepage mentions “Free access to DeepSeek-V3 and R1” deepseek.com. Possibly there’s a toggle or a drop-down to pick the model for a new chat. In the app, perhaps each new chat asks which model. This is great for customization: for brainstorming you might use V3, for logical problems R1.
- **Tool usage:** The toggles for DeepThink and Search effectively customize the behavior on a per-query basis.
- **User profiles:** There’s no mention of custom instructions yet, but because you can run it locally, one could in theory modify system prompts. The official UI hasn’t highlighted a “tell me about the user” feature like ChatGPT’s custom instructions.

That said, DeepSeek’s spirit is open-source and user-centric (the models are open or at least freely downloadable). They provide an **API and developer portal** larksuite.com, so if one wanted to integrate or tweak how the model responds, they could script around it.

In the app, one can also **upload documents or images as context** larksuite.com, which customizes that particular chat (giving it reference material). The interface note: “*Simply enter your prompt or upload documents or images to be used as references*” larksuite.com. This means you can feed it, say, a PDF report or an image (maybe a chart or diagram) and DeepSeek will use that in forming its answers. This is a direct customization of context per conversation, enabling highly tailored Q&A on user-provided data.

Integration with Business Tools & APIs: DeepSeek, being open/free, doesn’t natively integrate with proprietary business tools out-of-the-box, but it’s designed to be integrated by developers easily. They provide an **API** (with a free quota for basic usage) larksuite.com, which businesses can use to add DeepSeek’s capabilities into their apps or workflows. For instance, one could hook DeepSeek into a company Slack via a bot that calls the API. Or integrate it into an internal knowledge base search – given its strong reasoning, it could interpret queries and then search a database (especially if combined with the chain-of-thought approach where it can plan a search).

One integration aspect is **web search** which is built-in (so in a sense it integrates with search engines for you). When Search is enabled, DeepSeek will fetch real-time info. It also cites sources (likely with bracketed numbers linking to the references, similar to how Bing or

Perplexity does). This is valuable for tasks like market research or due diligence, and it keeps the user within the DeepSeek UI instead of having to switch to a browser manually.

Multi-platform support (web, desktop, mobile) means it integrates with your routine – you can have it on your phone as a companion or on your PC offline. They've essentially integrated the assistant *with the operating environment* to an extent – e.g., presumably the desktop app could be used alongside other programs, maybe via a global shortcut. If it's like some AI desktop apps, you might highlight text anywhere and press a key to ask DeepSeek about it (though that's speculative, but many local AI apps do that).

Multi-User Collaboration & Workspaces: DeepSeek's public offering is individual-centric. It doesn't have team accounts or shared project spaces in the UI. However, since the model can be self-hosted, a company could deploy a DeepSeek instance and make it available to multiple employees through a web interface or chat system. In that scenario, each user might either share one model instance (where each conversation is separate, but they technically are using the same backend) or have their own isolated sessions. DeepSeek doesn't manage identities beyond login on their official service.

One interesting note: DeepSeek's success and freakout reference [platformer.news](#) [platformer.news](#) indicates it got a lot of community attention. On a collaboration front, they might eventually open up features where people share interesting prompts or chain-of-thought examples. But nothing like that yet.

Accessibility: DeepSeek's explicit design choice to reveal chain-of-thought ironically doubles as an educational tool. It helps users learn *how to prompt better* by seeing how the AI interprets instructions [platformer.news](#) [platformer.news](#). This reduces the barrier for new users who are not sure how to use an AI effectively – they get guidance implicitly. It's a sort of built-in tutorial every time you ask something: the AI tells you "I think you meant this; here's how I will solve it." That's a unique form of accessibility – cognitive accessibility for understanding AI behavior.

The apps for DeepSeek likely incorporate features like **voice input** because Chinese AI developers often include voice. However, I haven't seen that explicitly. Even if not now, they could integrate an open speech recognition since it's local.

Language-wise, DeepSeek presumably supports English and Chinese strongly (founder is Chinese, likely bilingual training) [techtargget.com](#). It may support other languages too, but I'm not certain. The UI is in English (with a Chinese version of the site too). So they cater to a global audience.

For users with visual or motor impairments, the desktop app means you can operate it with standard accessibility tools (like using large font or high contrast). The chain-of-thought text is extra info but if it's distracting, there might be a setting to hide it if one prefers (not sure if they allow turning it off; they likely encourage it to stay on). If someone doesn't want the chain-of-thought, they can simply ignore that part of the text. But if needed, an update could allow toggling its visibility for a cleaner view.

Handling Long Documents, Code, and Tables: DeepSeek is capable of handling pretty complex tasks:

- **Long documents:** It encourages you to upload text and will summarize or QA on them larksuite.com. The context window of DeepSeek's models isn't explicitly stated, but likely at least 4k-8k tokens. If the doc is larger, the interface might use the Search function to find relevant parts (maybe that's partly what "DeepThink" does – it could chunk and reason across parts).
- **Code:** DeepSeek markets itself to coders as well. They even have a specialized model "DeepSeek Coder V2" and math models deepseek.com. Possibly the UI lets you choose those. If you ask coding questions, it might automatically route to the coder model. The UI displays code with proper formatting. Given it's a modern chat tool, it should have the standard code block formatting and copy button (though not confirmed, it's almost expected now in any new chat UI).
- One user scenario: DeepSeek was run locally by a developer with *Chatbox – a powerful client for AI models* reddit.com, meaning people integrate it with IDE plugins. If not the official UI, the existence of those options shows it can be used in various ways for coding. But in the official UI, you can paste code and ask for explanation or debugging. There is no built-in execution sandbox in the user-facing version, but a developer could pair DeepSeek with a local runtime for automated code testing.
- **Tables and data:** DeepSeek can certainly interpret and generate tables. It might not have an advanced dataviz tool, but if you give it CSV data, it can output insights or even generate Markdown tables in the answer. If DeepSeek's chain-of-thought is shown, you could see it thinking something like "to answer this, I should calculate the sum of column A" which is fascinating. However, heavy data like thousands of rows might overwhelm it unless it's allowed to use a tool (like Search mode might not help with local data, so this is where offline might need chunking).
- They explicitly list *"Summarizing Articles: Paste long articles or documents, and DeepSeek will provide concise summaries"* as a common use larksuite.com. And *"Quick Translations"* larksuite.com, so it's adept at language tasks too.

Security & Privacy: DeepSeek's slogan could be privacy. It's free, offline-capable, and open-source, implying a strong ethos of user control. *"Unlike many AI tools that require subscriptions or constant internet connectivity, DeepSeek is free to use and can operate offline, ensuring data privacy."* larksuite.com larksuite.com. This direct quote from their guide highlights privacy as a top feature. For businesses, being able to use an AI without sending data to an external server is a big deal. Even using the cloud version, since it's a Chinese company outside U.S. jurisdiction, some might find that ironically preferable (though others might have concerns the other way around – data on Chinese servers could be sensitive; but the company likely hosts globally available instances possibly not in China since they gained Western users).

DeepSeek's chain-of-thought approach was somewhat controversial because it might reveal internal reasoning that occasionally contains snippets of training data or biases. However, so far it's been seen as positive. They'll have to ensure the chain-of-thought doesn't inadvertently output something it shouldn't (e.g., sensitive info or harmful reasoning) – so far it seems fine.

From a security standpoint, if you run DeepSeek locally, you avoid network threats, but if you use their cloud, exercise normal caution. They have disclaimers (and the Platformer article suggests the world was freaked out partly because *an advanced reasoning model was offered for free by a Chinese firm, upending assumptions* platformer.news). But nothing indicates shady usage – likely they had a genuine community-driven approach.

Desktop vs Mobile: DeepSeek covers both well:

- **Desktop App:** They offer it on Windows/macOS/Linux (the “download” link suggests cross-platform) deepseek.com. This app likely bundles a local model or at least runs the UI offline while calling their API if online. The user guide shows even a QR code to get the app on mobile larksuite.com and states explicitly they have iOS/Android apps larksuite.com. The desktop app might allow running the model on your hardware if you have a capable GPU, or it might just be a wrapper for the web. Enthusiasts on Reddit mention running R1 locally via Ollama and using Chatbox UI reddit.com, which shows even third-party UI is possible.
- **Mobile App:** On mobile, the DeepSeek app gives you AI on the go. It reportedly supports voice input (given many Chinese AI apps do, though again not confirmed, but likely you can dictate and it will transcribe). Even if not, you can upload images from your camera or files stored on the phone for analysis, per the guide. Conversations likely sync if you log in, or they might remain local (the guide implies logging in to the app is needed larksuite.com, so presumably to sync across devices and record usage).
- The mobile UI is probably similar to ChatGPT’s: a clean chat with maybe an extra menu to toggle DeepThink/Search. Having an AI with chain-of-thought on mobile is quite novel – you could ask it to plan your day and it might say, “I see your calendar has meetings at 10 and 2 (if it had such access), I plan to schedule tasks around them.”

In sum, **DeepSeek’s UI** is cutting-edge in transparency and offers serious utility for free. It feels a bit like a community project (in spirit) that went mainstream, combining robust features (file upload, search integration, chain-of-thought view, offline mode) that appeal to both AI enthusiasts and business users looking for control and cost-effectiveness. The slight learning curve (understanding when to use DeepThink or how to interpret chain-of-thought) is offset by the richer understanding and trust it provides. As one analysis put it, *DeepSeek is not just an AI model story but a **design story*** platformer.news – showing how interface decisions (like revealing reasoning) can shape user experience significantly.

Now, with detailed looks at each of the seven AI tools’ interfaces, we can distill the key differences and similarities that matter for professional users. Below is a **comparison table** summarizing major UI features side-by-side, followed by a final analysis to help decision-makers choose the right interface for their needs.

Comparison Table: Key UI Features (ChatGPT vs Gemini vs Claude vs Poe vs Llama vs LeChat vs DeepSeek)

To provide a quick overview, the table below compares the seven AI chat interfaces across various feature dimensions relevant to business use:

Feature / Tool

ChatGPT (OpenAI)

Gemini (Google)

Claude (Anthropic)

Poe (Quora)

Llama Open-Source UIs (e.g. HuggingChat)

LeChat (Mistral AI)

DeepSeek

UI Design & Layout

Polished, minimalist chat; sidebar for chats & models.

Integrated across apps (chat in Gmail/Docs sidebars, mobile overlay); unified web chat with Google styling techcrunch.com techcrunch.com.

Clean chat interface; Projects sidebar for grouping chats/docs pymnts.com.

Multi-chat hub with model selection sidebar; slightly cluttered due to many options magai.co.

Resembles ChatGPT; simple chat layout, extra settings for models/tools belsterns.com.

Basic chat UI; model picker (Small/Large); few organizing features (beta) superannotate.com.

Chat interface with unique chain-of-thought display platformer.news; toggles for DeepThink & Search larksuite.com.

Ease of Use & Intuitiveness

Extremely user-friendly; no setup needed; plus mobile/desktop apps with voice techrepublic.com techrepublic.com.

Very easy for Google users; appears wherever you work (no switching apps); voice input "Hey Google" on mobile techcrunch.com venturebeat.com.

Straightforward Q&A; voice on mobile; advanced features (Projects, etc.) tucked away for those who need them venturebeat.com pymnts.com.

Easy start but many options; one interface for many models; cross-platform sync of chats allaboutai.com allaboutai.com.

Setup may require tech help (for self-hosting), but hosted versions (HuggingChat) are plug-and-play; highly customizable UI might confuse non-tech initially belsterns.com.

Simple to query, but some beta hiccups (e.g. no chat renaming yet) superannotate.com; free access lowers barrier.

Free and accessible; web, PC, and mobile apps larksuite.com; chain-of-thought helps users understand AI's reasoning platformer.news, though adds extra text to parse.

Conversation History & Prompt Management

Saved chats listed in sidebar; can rename chats superannotate.com; custom instructions for persistent context across chats.

All chats tied to Google acct; conversations sync across devices techcrunch.com; memory feature in Advanced plan stores user preferences and context techcrunch.com; can create "Gems" custom bots techcrunch.com.

Full history retention; can organize chats into **Projects** (shared context & docs) pymnts.com; custom instructions per Project pymnts.com; 100K+ token context for long memory pymnts.com.

Remembers chats per model; cross-device history sync allaboutai.com; can delete history allaboutai.com; supports multi-model chats (one thread, multiple AI replies) for comparing answers allaboutai.com.

History depends on the UI used: e.g. HuggingChat offers multi-conversation with login, Chatbot UI can save locally belsterns.com; users/teams can self-host to control data retention; system prompts editable for custom behavior.

History available but minimal management (no naming yet, basic list) superannotate.com; 32k token context for lengthy prior conversation continuity superannotate.com.

All past chats accessible in sidebar larksuite.com; each turn's chain-of-thought visible for auditing reasoning platformer.news; "New chat" resets context easily larksuite.com; supports file/image uploads as context larksuite.com.

Customization & Settings

Model switch (GPT-3.5/4); plugin enable/disable; theme toggle; custom instructions to set tone or info adventuresinai.com.

Choice of model tier (Gemini Pro vs Ultra) if subscribed en.wikipedia.org; "Deep Research" mode for multi-step answers techcrunch.com; user can create shareable **custom bots ("Gems")** with natural language instructions techcrunch.com.

Tone/length settings for replies; Project-level instructions (e.g. "use formal tone") pymnts.com; can attach documents to Projects for custom knowledge pymnts.com; voice options (different

voice styles on mobile) venturebeat.com.

Users can build **no-code custom bots** with personalized instructions allaboutai.com; community bot marketplace to find tailored bots allaboutai.com; toggle between dozens of models and even image generators allaboutai.com; paid plans for faster responses.

Fully customizable: can self-host and modify UI (branding, plugins) belsterns.com; supports multiple model backends (OpenAI, local LLaMA, etc.) belsterns.com; advanced users can tweak system prompts or integrate tools (e.g. web search, via open plugins) github.com.

Limited UI settings so far; user picks model size (Small/Large/Next) for each chat superannotate.com; no user-defined persona settings yet (besides what you prompt manually). Future updates expected to add more UI options.

DeepThink toggle lets user request more exhaustive reasoning larksuite.com; **Search** toggle pulls live web info with sources larksuite.com; model selection (DeepSeek V3 vs R1) available deepseek.com; offers API for custom integrations larksuite.com; can run offline (user controls data and model usage) larksuite.com larksuite.com.

Integration with External Tools

Via **Plugins** (e.g. web browser, databases, office apps) – enabled in UI for Plus users; ChatGPT Enterprise/Teams has API access and Slack plugin, but core UI is standalone chat.

Native Google Workspace integration: in Gmail, Docs, Sheets, etc. (side panel assists drafting/content) techcrunch.com; can act on personal data (emails, calendar) via voice commands venturebeat.com; **Google Search** integrated for fact-checking and real-time info en.wikipedia.org.

Slack integration (Claude bot in Slack channels); supports plug-ins via Anthropic partners; Claude can sync with Google Drive, etc., via its API; Projects allow internal knowledge integration (upload company docs) pymnts.com; Claude voice reads calendar/email for Pro users (integrated with Google services) venturebeat.com.

Integrates multiple AI services (OpenAI, Anthropic, etc.) under one UI; has an API for developers to embed Poe bots in other products linkedin.com; no direct plugins to business apps, but one can share Poe links externally.

Depends on chosen UI: HuggingChat adding tool use (web search, PDF reading) github.com; can be integrated by developers into any system (many use Llama via libraries in Slack, VS Code, etc.); fully open, so companies can build custom integrations (database query, CRM) directly into the UI code.

Currently no third-party integrations (works offline by design superannotate.com); planned features include web search and image generation modes digitrendz.blog; Mistral offers an API for the model on their platform, so integration is via the API rather than the UI.

Web search built-in (no plugin needed) larksuite.com; has **API** for custom app integration larksuite.com; desktop app might allow system-wide usage (e.g. highlight text and query DeepSeek); model can be self-hosted for intranet integration.

Collaboration & Team Features

ChatGPT Teams/Enterprise: shared “workspace” (managed accounts) adventuresinai.com, but no live multi-user chat; can share conversation links for viewing. Admin console for user management adventuresinai.com.

Works with Google’s collaboration – e.g. multiple users in a Doc can all invoke Gemini help; Meet summaries shared to all participants; no separate “Gemini team chat,” it augments existing collaborative apps. Admin controls via Google Workspace.

Claude for Teams: Projects can be shared among team members (shared chat feed & organizational knowledge base) pymnts.com pymnts.com; encourages discovering colleagues’ useful prompts pymnts.com. No simultaneous co-edit in one chat, but Slack integration allows group Q&A with Claude.

No team-specific features; each account is individual. Community aspect via shared bots, but that’s public. Data not siloed by org (unless self-hosted privately via API).

Multi-user depends on self-host setup: one can deploy a Llama UI on a server for multiple users, but must implement login/permissions. Some UIs (Chatbot UI) can be tweaked for multi-user. Generally no built-in team spaces or sharing – but one can export chat logs manually.

No multi-user or enterprise admin features yet; each user account stands alone. Mistral might introduce enterprise options later (the name “Enterprise AI assistant” suggests future plans) but currently collaboration means manually sharing answers outside the tool.

No built-in collab or team accounts on the public service (each login is separate); however, the openness means a company could host a DeepSeek server accessible to many. The chain-of-thought could be valuable in team settings for transparency. DeepSeek’s **Projects** concept not present – each session is user-specific.

Accessibility (UI & Features)

Voice input/output on mobile (and now on web for Plus) techrepublic.com; UI localized in multiple languages; high-contrast and screen-reader friendly design (simple HTML). Handles code, math with formatting for clarity.

Broad language support and **multilingual UI**; voice-centric interface on mobile (multiple voice personas) venturebeat.com; deeply integrated in common tools (lowers barrier for less tech-savvy – they access AI in familiar apps). Gemini can also convert text to visuals (Slides) aiding different communication needs techcrunch.com.

English-focused but also handles many languages; **voice mode** on mobile with transcripts and summaries for accessibility venturebeat.com; large context means can assist with very long texts (useful for users who need summaries of big documents instead of reading). Simple interface works with screen readers.

Supports input/output in many languages (explicit mention of Japanese, French, Spanish, etc.) allaboutai.com; mobile apps available (iOS/Android, plus a Mac app) for on-the-go use; interface might be cluttered for screen-readers due to multiple elements. No native voice, but can use device's speech-to-text if needed.

Varied: some UIs (like SpeechGPT) offer full voice conversation mode with multiple TTS/STT options belsterns.com; open UIs can be adapted for disabilities (e.g. one could integrate Braille display support or custom styling). Many open models support dozens of languages. Accessibility depends on the specific front-end chosen, but the open nature allows community improvements belsterns.com.

Multilingual (Mistral Large is polyglot) superannotate.com; straightforward UI (large text, few distractions); no built-in voice or screen-reader optimizations yet. But being web-based, it likely works in browser accessibility modes. The free aspect means anyone can try it without financial barrier.

Chain-of-thought transparency serves educational accessibility – users see how to better interact platformer.news; interface is relatively uncluttered aside from extra reasoning text. Mobile and desktop apps increase availability. Likely supports Chinese and English well (and maybe other languages to extent of training). No known voice feature built-in, but one could use OS dictation.

Handling Long Inputs (Docs, Code, Tables)

GPT-4 handles up to 32K tokens (Enterprise) adventuresincre.com; file uploads via Advanced Data Analysis (Code Interpreter) for beyond-context processing; outputs formatted tables and code with syntax highlighting. Great for code (even executes code in sandbox) and decent with large text via summarize tools.

Gemini Ultra can take large inputs (exact token limit not public, but designed for long conversations) techcrunch.com; can directly use files from Drive (no manual copy needed) techcrunch.com; in Sheets, it can work with live data to create tables/formulas techcrunch.com. Code execution possible in Advanced mode (Python in-chat) techcrunch.com; image analysis/generation supported.

200K token context in Claude 3.5 Projects (roughly 500 pages) pymnts.com – best for very long documents; allows **multiple file attachments** for reference pymnts.com; specialized “Artifacts” side-panel for large code outputs or diagrams pymnts.com. Excellent for processing big texts or codebases.

Underlying model limits (GPT-4 8K/32K, Claude 100K, etc.) apply; Poe itself doesn't extend context. No file upload feature in UI, so large docs must be pasted or summarized externally. Handles code well (will format it) but cannot run it. Multi-bot chat can help verify large output by cross-checking models.

Open-source models vary: LLaMA-2 default 4K context (some fine-tunes extend this); solutions like retrieval (via LlamaIndex) can handle book-size corpora by chunking and searching [belsterns.com](#). Many UIs allow **loading documents** into a vector database for Q&A. Code support: if using CodeLlama, outputs are high quality; user may integrate a code executor manually. Tables: will output Markdown tables, but for heavy data analysis, external tooling integration is needed (which one can wire up in custom UI).

32K token context window [superannotate.com](#) allows very long inputs in one go (dozens of pages); has **document analysis** built-in (upload PDF/text for Q&A) [mistral.ai](#). Good coding capabilities (Mistral model excels at code benchmarks [superannotate.com](#)) – outputs code with reasoning. No code execution environment in UI.

Can **upload images & documents** as context [larksuite.com](#) [larksuite.com](#); context length likely moderate (maybe 4-8K tokens), but "DeepThink" mode might effectively handle more by iterative analysis. Strong reasoning model (R1) can parse complex problems stepwise [platformer.news](#). Code output is supported (with proper formatting) and chain-of-thought will even explain code line-by-line if needed. Large data: would rely on its reasoning or search (no built-in DB integration yet unless user does via API).

Security & Privacy

Enterprise plan: SOC2 compliance, encryption, no training on your data [adventuresincre.com](#); Team plan domain verification and SSO [adventuresincre.com](#). User-side: can turn off chat logging [adventuresincre.com](#). However, base ChatGPT for individuals sends data to OpenAI's cloud (a consideration for highly sensitive info).

Data from Workspace usage not used to train models [en.wikipedia.org](#); Google's robust security (encryption, admin controls). Users benefit from Google's compliance and decades of security expertise. Some enterprises may worry about Google handling data, but terms for Duet/Gemini in Workspace assure privacy. On Pixel devices, Gemini is on-device for some features (no cloud needed for certain Assistant tasks).

Emphasizes privacy: team Projects data not used for training without consent [pymnts.com](#); encryption in transit/at rest; Claude can be deployed in isolated cloud (AWS Bedrock) for more control [aws.amazon.com](#). As a company, Anthropic positions itself as "safer" AI, with constitutional AI to avoid problematic outputs. Good track record so far, but fewer formal certs than OpenAI's enterprise (still new).

Quora stores chats and uses them to improve service (no training large models, but possibly to train their smaller models); you can delete data [allaboutai.com](#). No enterprise-grade isolation –

queries to GPT/Claude go through Poe's servers. Suited for less-sensitive usage. They do moderate content to prevent abuse.

Full data control if self-hosted – you decide what happens to conversation logs, and none need leave your network larksuite.com. Open models don't phone home. However, no out-of-box content filtering or compliance tools – you must implement any needed safety or logging. If using a community-hosted service (like HF Spaces), then trust and policies vary. Many open UIs allow turning off telemetry completely.

Mistral highlights European data compliance; running without internet means queries aren't hitting external APIs superannotate.com. However, using LeChat online means data is on Mistral's servers (hosted on Azure in EU likely) – they have a privacy policy and likely don't use chats to train without permission. As a newcomer, not yet certified for enterprise compliance publicly.

Very privacy-friendly: can run fully offline on your device larksuite.com. Even cloud use is free and not behind big corporate entity (though it's a startup). They openly released model weights (R1, etc.), showing commitment to open access platformer.news. One must trust the app if using cloud version, but since models can be local, businesses can choose to keep everything in-house. The chain-of-thought visibility also means the model is less of a black box – you can catch if it ever tries to do something odd.

DISCLAIMER

The information contained in this document is provided for educational and informational purposes only. We make no representations or warranties of any kind, express or implied, about the completeness, accuracy, reliability, suitability, or availability of the information contained herein.

Any reliance you place on such information is strictly at your own risk. In no event will IntuitionLabs.ai or its representatives be liable for any loss or damage including without limitation, indirect or consequential loss or damage, or any loss or damage whatsoever arising from the use of information presented in this document.

This document may contain content generated with the assistance of artificial intelligence technologies. AI-generated content may contain errors, omissions, or inaccuracies. Readers are advised to independently verify any critical information before acting upon it.

All product names, logos, brands, trademarks, and registered trademarks mentioned in this document are the property of their respective owners. All company, product, and service names used in this document are for identification purposes only. Use of these names, logos, trademarks, and brands does not imply endorsement by the respective trademark holders.

IntuitionLabs.ai is an AI software development company specializing in helping life-science companies implement and leverage artificial intelligence solutions. Founded in 2023 by [Adrien Laurent](#) and based in San Jose, California.

This document does not constitute professional or legal advice. For specific guidance related to your business needs, please consult with appropriate qualified professionals.

© 2025 IntuitionLabs.ai. All rights reserved.