

# Custom AI Studio: Local Gemini Interface Implementation

Based on your requirements and the existing `app.py`, I can design a custom UI that mimics Google AI Studio's features while providing local chat storage with cloud options. Here's a complete implementation plan leveraging your existing Gemini API integration:

## Backend Enhancements

Let's extend your `app.py` with essential features for a Google AI Studio-like experience:

```
# Add to app.py

# Storage strategy classes
class StorageBackend:
    def save_history(self, history_data): pass
    def load_history(self): pass
    def clear_history(self): pass

class FileSystemStorage(StorageBackend):
    def __init__(self, file_path):
        self.file_path = file_path

    def save_history(self, history_data):
        with open(self.file_path, "w", encoding='utf-8') as f:
            f.write(history_data)

    def load_history(self):
        if os.path.exists(self.file_path):
            with open(self.file_path, "r", encoding='utf-8') as f:
                return f.read()
        return ""

class CloudStorage(StorageBackend):
    """Cloud storage implementation (Firebase, S3, etc)"""
    # Implementation depends on your cloud provider
    pass

# Summary caching system
def summarize_conversation(history, max_length=1000):
    """Generate a summary of the conversation using Gemini"""
    if not history:
        return ""

    summary_prompt = "Create a concise summary of this conversation:\n\n"
    conversation_text = format_history_text(history)
    summary_prompt += conversation_text
```

```

try:
    summary = call_gemini_sync(summary_prompt, model_to_use='gemini-1.5-flash')
    return summary
except Exception as e:
    logger.error(f"Error summarizing conversation: {e}")
    return "Error generating summary"

# Streaming endpoint
@app.route('/generate_stream', methods=['POST'])
def generate_stream():
    """Stream generation results as they become available"""
    try:
        data = request.get_json()
        prompt = data.get('prompt', '')
        selected_model_id = data.get('model', 'gemini-1.5-flash')

        if not prompt:
            return jsonify({"error": "No text prompt provided"}), 400

        # Create model and configure for streaming
        model_name = f'models/{selected_model_id}'
        model = genai.GenerativeModel(model_name)

        def generate():
            yield "data: {\n\"status\": \"connected\"\n\n}"

            current_structured_history = load_structured_history()
            response = model.generate_content([prompt], stream=True)

            complete_response = ""

            for chunk in response:
                if hasattr(chunk, 'text') and chunk.text:
                    chunk_text = chunk.text
                    complete_response += chunk_text
                    yield f"data: {json.dumps({'chunk': chunk_text})}\n\n"

            updated_history = current_structured_history + [
                {"role": "user", "parts": [prompt]},
                {"role": "model", "parts": [complete_response]}
            ]
            save_structured_history(updated_history)

            yield f"data: {json.dumps({'status': 'complete'})}\n\n"

        return Response(generate(), mimetype='text/event-stream')

    except Exception as e:
        logger.error(f"Error in streaming generation: {e}")
        return jsonify({"error": f"{str(e)}"}), 500

# Token counting endpoint
@app.route('/count_tokens', methods=['POST'])
def count_tokens_endpoint():
    """Count tokens for the given text"""
    try:

```

```

data = request.get_json()
text = data.get('text', '')
model_id = data.get('model', 'gemini-1.5-flash')

model = genai.GenerativeModel(f'models/{model_id}')
token_count = model.count_tokens(text)

return jsonify({"tokens": token_count.total_tokens})
except Exception as e:
    logger.error(f"Error counting tokens: {e}")
    return jsonify({"error": str(e)}), 500

```

## Frontend Implementation

Now, here's the HTML template that mimics Google AI Studio:

```

<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Local AI Studio</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='css/main.css') }}">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.7.0/highlight.min.js">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css">
    <script src="https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.7.0/highlight.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/marked/4.0.0/marked.min.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
</head>
<body class="light-theme">
    <div>

        <aside class="sidebar">
            <div>
                <h1>Local AI Studio</h1>
                <button id="new-chat-btn" class="primary-button">
                    <i class="fas fa-plus"></i> New Chat
                </button>
            </div>

            <div>
                <h2>Conversations</h2>
                <div>
                    <div></div>
                </div>
            </div>

            <div>
                <h2>Settings</h2>
                <div>
                    <label for="theme-toggle">Dark Mode</label>
                    <div>
                        <input type="checkbox" id="theme-toggle">
                        <label for="theme-toggle"></label>
                    </div>
                </div>
            </div>
        </aside>
    </div>

```

```

        </div>
    </div>

    <div>
        <label for="storage-location">Storage Location</label>
        <select id="storage-location" class="select-dropdown">
            <option value="local">Local Storage</option>
            <option value="cloud">Cloud Storage</option>
        </select>
    </div>

    <button id="clear-history-btn" class="secondary-button">
        <i class="fas fa-trash"></i> Clear History
    </button>
</div>
</aside>

<main class="main-content">
    <div>
        <div>
            <label for="model-select">Model:</label>
            <select id="model-select" class="select-dropdown">
                {% for model in models %}
                    <option value="{{ model.id }}">{{ model.name }}</option>
                {% endfor %}
            </select>
        </div>

        <div>
            <div>
                <label for="temperature-slider">Temperature:</label>
                <input type="range" id="temperature-slider" min="0" max="1" step=
                <span>0.7</span>
            </div>

            <div>
                <label for="max-tokens">Max Output Tokens:</label>
                <input type="number" id="max-tokens" min="1" max="8192" value=
            </div>
        </div>
    </div>

    <div>
        <div>

        </div>

        <div>
            <button id="show-system-instructions" class="text-button">
                <i class="fas fa-cog"></i> System Instructions
            </button>
            <div>
                <textarea id="system-instructions" placeholder="You are a help
                <button id="save-instructions" class="primary-button">Save
            </div>
        </div>
    </div>

```

```

    </div>

    <div>
      <div>0 tokens</div>
      <div>
        <input type="text" id="prompt-input" placeholder="Type your message here" />
        <div>
          <button id="upload-image-btn" class="icon-button" title="Upload Image" >
            <i class="fas fa-image"></i>
          </button>
          <button id="upload-audio-btn" class="icon-button" title="Upload Audio" >
            <i class="fas fa-microphone"></i>
          </button>
          <button id="upload-video-btn" class="icon-button" title="Upload Video" >
            <i class="fas fa-video"></i>
          </button>
          <button id="upload-document-btn" class="icon-button" title="Upload Document" >
            <i class="fas fa-file-alt"></i>
          </button>
          <button id="send-btn" class="primary-button" >
            <i class="fas fa-paper-plane"></i>
          </button>
        </div>
      </div>
    </div>
  </div>

  <div>
    <div>
      <div>
        <div>
          <span>Input Tokens:</span>
          <span>0</span>
        </div>
        <div>
          <span>Output Tokens:</span>
          <span>0</span>
        </div>
        <div>
          <span>Total Tokens:</span>
          <span>0</span>
        </div>
      </div>
    </div>
  </div>
</main>

<aside class="context-sidebar hidden" id="context-sidebar">
  <div>
    <h2>Context</h2>
    <button id="close-context-btn" class="icon-button">
      <i class="fas fa-times"></i>
    </button>
  </div>

  <div>

```

```

        <h3>Conversation Summary</h3>
        <div>

        </div>
    </div>

    <div>
        <h3>Uploaded Files</h3>
        <div>

        </div>
    </div>
    &lt;/aside&gt;
</div>

<div>
    <div>
        <div>
            <h2>Upload File</h2>
            &lt;button id="close-upload-modal" class="icon-button"&gt;
                &lt;i class="fas fa-times"&gt;&lt;/i&gt;
            &lt;/button&gt;
        </div>
        <div>
            &lt;form id="upload-form" enctype="multipart/form-data"&gt;
                <div>
                    &lt;label for="file-input"&gt;Select File:&lt;/label&gt;
                    &lt;input type="file" id="file-input" name="file"&gt;
                </div>
                <div>
                    &lt;label for="file-description"&gt;Description (optional):&lt;/label&gt;
                    &lt;input type="text" id="file-description" name="description" placeholder="Description of the file"&gt;
                </div>
                &lt;button type="submit" class="primary-button"&gt;Upload&lt;/button&gt;
            &lt;/form&gt;
        </div>
    </div>
</div>

    &lt;script src="{% url_for('static', filename='js/main.js') %}"&gt;&lt;/script&gt;
&lt;/body&gt;
&lt;/html&gt;

```

## CSS Styling

Create this CSS file to match Google AI Studio's appearance:

```

/* static/css/main.css */
:root {
    /* Light Theme */
    --light-bg: #ffffff;
    --light-sidebar-bg: #f5f5f5;
    --light-text: #333333;
    --light-border: #e0e0e0;
}

```

```
--light-hover: #f0f0f0;
--light-primary: #1a73e8;
--light-secondary: #5f6368;
--light-user-message-bg: #f0f7ff;
--light-bot-message-bg: #f9f9f9;

/* Dark Theme */
--dark-bg: #202124;
--dark-sidebar-bg: #292a2d;
--dark-text: #e8eaed;
--dark-border: #5f6368;
--dark-hover: #35363a;
--dark-primary: #8ab4f8;
--dark-secondary: #9aa0a6;
--dark-user-message-bg: #31343a;
--dark-bot-message-bg: #3c4043;
}

/* Base Styles */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  font-family: 'Google Sans', 'Roboto', Arial, sans-serif;
  font-size: 16px;
  line-height: 1.5;
  transition: background-color 0.3s, color 0.3s;
  height: 100vh;
  overflow: hidden;
}

/* Theme Classes */
.light-theme {
  background-color: var(--light-bg);
  color: var(--light-text);
}

.dark-theme {
  background-color: var(--dark-bg);
  color: var(--dark-text);
}

/* App Container */
.app-container {
  display: flex;
  height: 100vh;
  width: 100vw;
}

/* Sidebar Styles */
.sidebar {
  width: 280px;
  height: 100%;
```

```
padding: 16px;
display: flex;
flex-direction: column;
overflow-y: auto;
}

.light-theme .sidebar {
background-color: var(--light-sidebar-bg);
border-right: 1px solid var(--light-border);
}

.dark-theme .sidebar {
background-color: var(--dark-sidebar-bg);
border-right: 1px solid var(--dark-border);
}

.sidebar-header {
padding-bottom: 16px;
margin-bottom: 16px;
border-bottom: 1px solid;
}

.light-theme .sidebar-header {
border-color: var(--light-border);
}

.dark-theme .sidebar-header {
border-color: var(--dark-border);
}

.sidebar-header h1 {
font-size: 20px;
font-weight: 500;
margin-bottom: 16px;
}

/* Button Styles */
.primary-button {
background-color: var(--light-primary);
color: white;
border: none;
padding: 8px 16px;
border-radius: 4px;
cursor: pointer;
font-weight: 500;
transition: background-color 0.2s;
}

.dark-theme .primary-button {
background-color: var(--dark-primary);
}

.primary-button:hover {
background-color: #1765cc;
}
```



```
.dark-theme .primary-button:hover {
  background-color: #6fa8f5;
}

/* Chat Container */
.main-content {
  flex: 1;
  display: flex;
  flex-direction: column;
  height: 100%;
  overflow: hidden;
}

.model-header {
  padding: 16px;
  border-bottom: 1px solid;
  display: flex;
  justify-content: space-between;
  align-items: center;
}

.light-theme .model-header {
  border-color: var(--light-border);
}

.dark-theme .model-header {
  border-color: var(--dark-border);
}

.chat-container {
  flex: 1;
  display: flex;
  flex-direction: column;
  overflow: hidden;
}

.chat-messages {
  flex: 1;
  overflow-y: auto;
  padding: 16px;
}

.message {
  margin-bottom: 16px;
  max-width: 85%;
  padding: 12px 16px;
  border-radius: 8px;
  line-height: 1.5;
}

.user-message {
  margin-left: auto;
  border-radius: 8px 8px 0 8px;
}

.light-theme .user-message {
```

```
        background-color: var(--light-user-message-bg);
    }

    .dark-theme .user-message {
        background-color: var(--dark-user-message-bg);
    }

    .bot-message {
        margin-right: auto;
        border-radius: 0 8px 8px 8px;
    }

    .light-theme .bot-message {
        background-color: var(--light-bot-message-bg);
    }

    .dark-theme .bot-message {
        background-color: var(--dark-bot-message-bg);
    }

    /* Input Area */
    .input-area {
        padding: 16px;
        border-top: 1px solid;
    }

    .light-theme .input-area {
        border-color: var(--light-border);
    }

    .dark-theme .input-area {
        border-color: var(--dark-border);
    }

    .token-counter {
        text-align: right;
        font-size: 12px;
        margin-bottom: 4px;
        color: var(--light-secondary);
    }

    .dark-theme .token-counter {
        color: var(--dark-secondary);
    }

    .input-container {
        position: relative;
    }

    .input-container textarea {
        width: 100%;
        min-height: 60px;
        padding: 12px;
        border-radius: 8px;
        resize: none;
        font-family: inherit;
```

```

        font-size: inherit;
        transition: border 0.3s;
    }

    .light-theme .input-container textarea {
        background-color: var(--light-bg);
        border: 1px solid var(--light-border);
        color: var(--light-text);
    }

    .dark-theme .input-container textarea {
        background-color: var(--dark-bg);
        border: 1px solid var(--dark-border);
        color: var(--dark-text);
    }

    .input-buttons {
        display: flex;
        align-items: center;
        margin-top: 8px;
    }

    .icon-button {
        background: none;
        border: none;
        width: 36px;
        height: 36px;
        border-radius: 50%;
        display: flex;
        align-items: center;
        justify-content: center;
        cursor: pointer;
        margin-right: 8px;
        transition: background-color 0.2s;
    }

    .light-theme .icon-button {
        color: var(--light-secondary);
    }

    .dark-theme .icon-button {
        color: var(--dark-secondary);
    }

    .light-theme .icon-button:hover {
        background-color: var(--light-hover);
    }

    .dark-theme .icon-button:hover {
        background-color: var(--dark-hover);
    }

    /* Model Controls */
    .model-selector, .generation-controls {
        display: flex;
        align-items: center;
    }

```

```

        gap: 12px;
    }

    .select-dropdown {
        padding: 8px 12px;
        border-radius: 4px;
        font-family: inherit;
    }

    .light-theme .select-dropdown {
        background-color: var(--light-bg);
        border: 1px solid var(--light-border);
        color: var(--light-text);
    }

    .dark-theme .select-dropdown {
        background-color: var(--dark-bg);
        border: 1px solid var(--dark-border);
        color: var(--dark-text);
    }

    /* Utility Classes */
    .hidden {
        display: none !important;
    }

    /* Code Blocks */
    pre {
        margin: 16px 0;
        border-radius: 4px;
        overflow-x: auto;
    }

    code {
        font-family: 'Roboto Mono', monospace;
    }

    /* Responsive Design */
    @media (max-width: 768px) {
        .app-container {
            flex-direction: column;
        }

        .sidebar {
            width: 100%;
            height: auto;
            max-height: 30vh;
        }

        .model-header {
            flex-direction: column;
            align-items: flex-start;
        }

        .generation-controls {
            margin-top: 12px;
        }
    }

```

```

        width: 100%;
    }
}

```

## JavaScript Implementation

Finally, here's the JavaScript that powers the interface:

```

// static/js/main.js
document.addEventListener('DOMContentLoaded', function() {
    // Elements
    const themeToggle = document.getElementById('theme-toggle');
    const modelSelect = document.getElementById('model-select');
    const temperatureSlider = document.getElementById('temperature-slider');
    const temperatureValue = document.getElementById('temperature-value');
    const maxTokens = document.getElementById('max-tokens');
    const chatMessages = document.getElementById('chat-messages');
    const promptInput = document.getElementById('prompt-input');
    const tokenCounter = document.getElementById('token-counter');
    const sendBtn = document.getElementById('send-btn');
    const uploadImageBtn = document.getElementById('upload-image-btn');
    const uploadAudioBtn = document.getElementById('upload-audio-btn');
    const uploadVideoBtn = document.getElementById('upload-video-btn');
    const uploadDocumentBtn = document.getElementById('upload-document-btn');
    const uploadModal = document.getElementById('upload-modal');
    const closeUploadModalBtn = document.getElementById('close-upload-modal');
    const uploadForm = document.getElementById('upload-form');

    // State
    let currentModelId = modelSelect.value;
    let currentTheme = localStorage.getItem('theme') || 'light';
    let uploadType = null;
    let tokenUsage = { input: 0, output: 0 };

    // Theme management
    function applyTheme(theme) {
        if (theme === 'dark') {
            document.body.classList.remove('light-theme');
            document.body.classList.add('dark-theme');
            themeToggle.checked = true;
        } else {
            document.body.classList.add('light-theme');
            document.body.classList.remove('dark-theme');
            themeToggle.checked = false;
        }
        localStorage.setItem('theme', theme);
    }

    <div>❌</div>

    [^1]: https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/62143253/aa1a38d4-0f31
    [^2]: https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/62143253/606057d9-023c
    [^3]: https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/62143253/83e97c96-f136
    [^4]: https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/62143253/7730dfad-65ac
    [^5]: https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/62143253/af37951c-4d39
    [^6]: https://pplx-res.cloudinary.com/image/upload/v1743740742/user_uploads/cayiJwreAuEke

```

[^7]: [https://www.reddit.com/r/Bard/comments/1dzvh4l/why\\_would\\_i\\_choose\\_the\\_gemini\\_website/](https://www.reddit.com/r/Bard/comments/1dzvh4l/why_would_i_choose_the_gemini_website/)

[^8]: <https://www.youtube.com/watch?v=6h9y1rLem4c>

[^9]: [https://www.reddit.com/r/selfhosted/comments/1j2q0lp/best\\_tools\\_strategies\\_for\\_full\\_stack\\_development/](https://www.reddit.com/r/selfhosted/comments/1j2q0lp/best_tools_strategies_for_full_stack_development/)

[^10]: <https://lablab.ai/t/google-ai-studio>

[^11]: [https://www.reddit.com/r/WritingWithAI/comments/1cxajvj/no\\_cloud\\_ai\\_tools/](https://www.reddit.com/r/WritingWithAI/comments/1cxajvj/no_cloud_ai_tools/)

[^12]: [https://www.reddit.com/r/LangChain/comments/1bh8o17/does\\_it\\_make\\_sense\\_to\\_store\\_chat\\_history\\_in\\_a\\_database/](https://www.reddit.com/r/LangChain/comments/1bh8o17/does_it_make_sense_to_store_chat_history_in_a_database/)

[^13]: [https://www.reddit.com/r/webdev/comments/1ao3357/storing\\_conversation\\_in\\_local\\_storage/](https://www.reddit.com/r/webdev/comments/1ao3357/storing_conversation_in_local_storage/)

[^14]: <https://klu.ai/glossary/google-ai-studio>

[^15]: [https://www.reddit.com/r/selfhosted/comments/1jbk06h/the\\_complete\\_guide\\_to\\_building\\_a\\_self\\_hosted\\_llm\\_application/](https://www.reddit.com/r/selfhosted/comments/1jbk06h/the_complete_guide_to_building_a_self_hosted_llm_application/)

[^16]: <https://ai.google.dev/gemini-api/docs/ai-studio-quickstart>

[^17]: [https://www.reddit.com/r/selfhosted/comments/1hg8q1q/a\\_local\\_application\\_for\\_storing\\_chat\\_history/](https://www.reddit.com/r/selfhosted/comments/1hg8q1q/a_local_application_for_storing_chat_history/)

[^18]: <https://magai.co/how-to-organize-ai-chat-history/>

[^19]: <https://www.datacamp.com/tutorial/local-ai>

[^20]: <https://www.semanticscholar.org/paper/96ba4868e061187ca2f384833c3c7484c207a6d9>

[^21]: <https://www.semanticscholar.org/paper/5f54cc987cac2f456e917f65f0cdb9547c16a98>

[^22]: <https://www.semanticscholar.org/paper/d62af5878187344ac7479819e014d64c15e8cf92>

[^23]: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11436130/>

[^24]: [https://www.reddit.com/r/Bard/comments/1daevii/why\\_google\\_ai\\_studio\\_is\\_free\\_when\\_gemini\\_is\\_not/](https://www.reddit.com/r/Bard/comments/1daevii/why_google_ai_studio_is_free_when_gemini_is_not/)

[^25]: [https://www.reddit.com/r/Bard/comments/1joonz5/google\\_ai\\_studio\\_is\\_unusable\\_past\\_5\\_minutes/](https://www.reddit.com/r/Bard/comments/1joonz5/google_ai_studio_is_unusable_past_5_minutes/)

[^26]: [https://www.reddit.com/r/singularity/comments/1hcrmbg/google\\_gemini\\_20\\_realtime\\_ai\\_chat/](https://www.reddit.com/r/singularity/comments/1hcrmbg/google_gemini_20_realtime_ai_chat/)

[^27]: [https://www.reddit.com/r/Bard/comments/1hlj29t/having\\_gotten\\_used\\_to\\_using\\_models\\_from\\_google\\_ai\\_studio/](https://www.reddit.com/r/Bard/comments/1hlj29t/having_gotten_used_to_using_models_from_google_ai_studio/)

[^28]: [https://www.reddit.com/r/GeminiAI/comments/1jkjvy9/better\\_ui\\_for\\_google\\_ai\\_studio/](https://www.reddit.com/r/GeminiAI/comments/1jkjvy9/better_ui_for_google_ai_studio/)

[^29]: [https://www.reddit.com/r/Bard/comments/1f92t62/what\\_is\\_function\\_calling\\_in\\_ai\\_studio/](https://www.reddit.com/r/Bard/comments/1f92t62/what_is_function_calling_in_ai_studio/)

[^30]: [https://www.reddit.com/r/Bard/comments/1if3iik/whats\\_the\\_ai\\_studio/](https://www.reddit.com/r/Bard/comments/1if3iik/whats_the_ai_studio/)

[^31]: [https://www.reddit.com/r/Bard/comments/1jlyzsq/i\\_redesigned\\_google\\_ai\\_studio\\_using\\_figma/](https://www.reddit.com/r/Bard/comments/1jlyzsq/i_redesigned_google_ai_studio_using_figma/)

[^32]: [https://www.reddit.com/r/Bard/comments/1lrh98p/newbie\\_here\\_and\\_playing\\_with\\_google\\_ai\\_studio/](https://www.reddit.com/r/Bard/comments/1lrh98p/newbie_here_and_playing_with_google_ai_studio/)

[^33]: [https://www.reddit.com/r/singularity/comments/1bm2hp1/those\\_of\\_us\\_who\\_now\\_how\\_to\\_develop\\_with\\_google\\_ai\\_studio/](https://www.reddit.com/r/singularity/comments/1bm2hp1/those_of_us_who_now_how_to_develop_with_google_ai_studio/)

[^34]: [https://www.reddit.com/r/MicrosoftEdge/comments/1jn6j7n/ui\\_issue\\_in\\_google\\_studio/](https://www.reddit.com/r/MicrosoftEdge/comments/1jn6j7n/ui_issue_in_google_studio/)

[^35]: [https://www.reddit.com/r/Bard/comments/1jjsmok/gemini\\_25\\_pro\\_feels\\_illegal\\_to\\_use\\_in\\_us/](https://www.reddit.com/r/Bard/comments/1jjsmok/gemini_25_pro_feels_illegal_to_use_in_us/)

[^36]: [https://www.reddit.com/r/Bard/comments/1jm9m5o/increased\\_limits\\_new\\_features\\_in\\_ai\\_studio/](https://www.reddit.com/r/Bard/comments/1jm9m5o/increased_limits_new_features_in_ai_studio/)

[^37]: [https://www.reddit.com/r/Bard/comments/1g6fhis/alternatives\\_to\\_google\\_ai\\_studio\\_for\\_local\\_development/](https://www.reddit.com/r/Bard/comments/1g6fhis/alternatives_to_google_ai_studio_for_local_development/)

[^38]: <https://www.youtube.com/watch?v=XcSgF2zdeT8>

[^39]: <https://cloud.google.com/ai/generative-ai>

[^40]: <https://cloud.google.com/ai/apis>

[^41]: <https://ai.google/get-started/for-developers/>

[^42]: <https://cloud.google.com/generative-ai-studio>

[^43]: <https://www.youtube.com/watch?v=4oyqd7CB09c>

[^44]: <https://aistudio.google.com>

[^45]: <https://developers.googleblog.com/en/gemini-api-and-ai-studio-now-offer-grounding/>

[^46]: <https://ai.google/get-started/products/>

[^47]: <https://www.youtube.com/watch?v=IH0JUJjZbzc>

[^48]: <https://ai.google.dev/aistudio>

[^49]: [https://www.reddit.com/r/LLMDevs/comments/1f3vhw0/llms\\_in\\_the\\_cloud\\_vs\\_running\\_locally/](https://www.reddit.com/r/LLMDevs/comments/1f3vhw0/llms_in_the_cloud_vs_running_locally/)

[^50]: [https://www.reddit.com/r/homelab/comments/1cp3rqc/local\\_nas\\_with\\_ai/](https://www.reddit.com/r/homelab/comments/1cp3rqc/local_nas_with_ai/)

[^51]: [https://www.reddit.com/r/selfhosted/comments/1i6ggyh/got\\_deepseek\\_r1\\_running\\_locally/](https://www.reddit.com/r/selfhosted/comments/1i6ggyh/got_deepseek_r1_running_locally/)

[^52]: [https://www.reddit.com/r/ollama/comments/1hwe7cg/databridge\\_opensource\\_local\\_multi\\_llm\\_chat\\_interface/](https://www.reddit.com/r/ollama/comments/1hwe7cg/databridge_opensource_local_multi_llm_chat_interface/)

[^53]: [https://www.reddit.com/r/homelab/comments/17dgzme/does\\_anyone\\_in\\_here\\_run\\_ai\\_stuff\\_locally/](https://www.reddit.com/r/homelab/comments/17dgzme/does_anyone_in_here_run_ai_stuff_locally/)

[^54]: [https://www.reddit.com/r/softwarearchitecture/comments/1dk5dn5/ai\\_tools\\_you\\_use\\_as\\_a\\_developer/](https://www.reddit.com/r/softwarearchitecture/comments/1dk5dn5/ai_tools_you_use_as_a_developer/)

[^55]: [https://www.reddit.com/r/sysadmin/comments/18aczys/what\\_are\\_people\\_using\\_for\\_onpremise\\_monitoring/](https://www.reddit.com/r/sysadmin/comments/18aczys/what_are_people_using_for_onpremise_monitoring/)

[^56]: [https://www.reddit.com/r/mlops/comments/1fe5i82/how\\_to\\_get\\_started\\_with\\_building\\_a\\_ml\\_pipeline/](https://www.reddit.com/r/mlops/comments/1fe5i82/how_to_get_started_with_building_a_ml_pipeline/)

[^57]: [https://www.reddit.com/r/LocalLLaMA/comments/1chk162/langchain\\_vs\\_llama\\_index\\_vs\\_cohere/](https://www.reddit.com/r/LocalLLaMA/comments/1chk162/langchain_vs_llama_index_vs_cohere/)

[^58]: [https://www.reddit.com/r/cloudcomputing/comments/1hbmvw7/whats\\_the\\_future\\_of\\_multi\\_model\\_development/](https://www.reddit.com/r/cloudcomputing/comments/1hbmvw7/whats_the_future_of_multi_model_development/)

[^59]: [https://www.reddit.com/r/selfhosted/comments/1b0wqgu/building\\_my\\_own\\_ai\\_server\\_for\\_home/](https://www.reddit.com/r/selfhosted/comments/1b0wqgu/building_my_own_ai_server_for_home/)

[^60]: [https://www.reddit.com/r/LocalLLaMA/comments/1imy754/update\\_building\\_a\\_fully\\_opensource\\_llm\\_server/](https://www.reddit.com/r/LocalLLaMA/comments/1imy754/update_building_a_fully_opensource_llm_server/)

[^61]: [https://www.reddit.com/r/aws/comments/mr6jnv/does\\_aws\\_have\\_cloud\\_storage\\_for\\_simple\\_data\\_storage/](https://www.reddit.com/r/aws/comments/mr6jnv/does_aws_have_cloud_storage_for_simple_data_storage/)

[^62]: [https://www.reddit.com/r/LocalLLaMA/comments/18dxub5/looking\\_to\\_build\\_a\\_new\\_system\\_architecture/](https://www.reddit.com/r/LocalLLaMA/comments/18dxub5/looking_to_build_a_new_system_architecture/)

[^63]: <https://cloud.google.com/use-cases/multimodal-ai>  
[^64]: <https://www.webai.com/blog/cloud-ai-vs-local-ai-which-is-best-for-your-business>  
[^65]: <https://cloud.google.com/storage/docs/storage-classes>  
[^66]: <https://www.youtube.com/watch?v=WKj8lo-ZbY4>  
[^67]: <https://cloud.google.com/architecture/ai-ml/storage-for-ai-ml>  
[^68]: <https://cloud.google.com/storage>  
[^69]: <https://www.pcmag.com/how-to/how-to-run-your-own-chatgpt-like-llm-for-free-and-in>  
[^70]: <https://ai.google/get-started/our-models/>  
[^71]: <https://bizbot.com/blog/cloud-storage-10-selection-criteria/>  
[^72]: <https://www.youtube.com/watch?v=WjrdR0NU4Sk>  
[^73]: <https://modal.com>  
[^74]: <https://www.gov.uk/guidance/meeting-digital-and-technology-standards-in-schools-ar>  
[^75]: <https://cloud.google.com/use-cases/free-ai-tools>  
[^76]: [https://www.reddit.com/r/LocalLLaMA/comments/1i1ffid/i\\_accidentally\\_built\\_an\\_open\\_](https://www.reddit.com/r/LocalLLaMA/comments/1i1ffid/i_accidentally_built_an_open_)  
[^77]: [https://www.reddit.com/r/FlutterDev/comments/1i5mf0q/are\\_there\\_any\\_ai\\_tools\\_for\\_ui](https://www.reddit.com/r/FlutterDev/comments/1i5mf0q/are_there_any_ai_tools_for_ui)  
[^78]: [https://www.reddit.com/r/OpenAI/comments/1hd2r2b/gemini\\_20\\_is\\_what\\_4o\\_was\\_supposed](https://www.reddit.com/r/OpenAI/comments/1hd2r2b/gemini_20_is_what_4o_was_supposed)  
[^79]: [https://www.reddit.com/r/LocalLLaMA/comments/1heqo18/a\\_functional\\_nicelooking\\_web\\_](https://www.reddit.com/r/LocalLLaMA/comments/1heqo18/a_functional_nicelooking_web_)  
[^80]: [https://www.reddit.com/r/ChatGPTCoding/comments/1excgal/whats\\_the\\_best\\_ai\\_tool\\_to\\_](https://www.reddit.com/r/ChatGPTCoding/comments/1excgal/whats_the_best_ai_tool_to_)  
[^81]: [https://www.reddit.com/r/LocalLLaMA/comments/1jnx49/i\\_made\\_a\\_grammarly\\_alternativ](https://www.reddit.com/r/LocalLLaMA/comments/1jnx49/i_made_a_grammarly_alternativ)  
[^82]: [https://www.reddit.com/r/Bard/comments/1jlr0f/gemini\\_ai\\_studios\\_unremovable\\_ui\\_ey](https://www.reddit.com/r/Bard/comments/1jlr0f/gemini_ai_studios_unremovable_ui_ey)  
[^83]: [https://www.reddit.com/r/web\\_design/comments/1ah8pd9/are\\_there\\_any\\_free\\_ai\\_tools\\_a](https://www.reddit.com/r/web_design/comments/1ah8pd9/are_there_any_free_ai_tools_a)  
[^84]: [https://www.reddit.com/r/singularity/comments/1hbwcci/examples\\_of\\_native\\_image\\_out](https://www.reddit.com/r/singularity/comments/1hbwcci/examples_of_native_image_out)  
[^85]: [https://www.reddit.com/r/softwaredevelopment/comments/11n0ibu/how\\_start\\_using\\_ai\\_i](https://www.reddit.com/r/softwaredevelopment/comments/11n0ibu/how_start_using_ai_i)  
[^86]: [https://www.reddit.com/r/LocalLLaMA/comments/1fdkegs/best\\_modelapplication\\_to\\_rep](https://www.reddit.com/r/LocalLLaMA/comments/1fdkegs/best_modelapplication_to_rep)  
[^87]: [https://www.reddit.com/r/Bard/comments/1epinjq/best\\_gui\\_to\\_use\\_with\\_google\\_studio\\_](https://www.reddit.com/r/Bard/comments/1epinjq/best_gui_to_use_with_google_studio_)  
[^88]: [https://www.reddit.com/r/graphic\\_design/comments/16rxm5g/the\\_15\\_most\\_useful\\_free\\_t](https://www.reddit.com/r/graphic_design/comments/16rxm5g/the_15_most_useful_free_t)  
[^89]: <https://slashdot.org/software/p/Google-AI-Studio/alternatives>  
[^90]: <https://www.linkedin.com/advice/3/how-do-you-design-user-friendly-ai-interfaces>  
[^91]: <https://www.youtube.com/watch?v=GX9wrTxs2ow>  
[^92]: <https://uizard.io>  
[^93]: <https://www.pragmaticcoders.com/blog/designing-ai-interfaces-challenges-trends-and>  
[^94]: <https://www.youtube.com/watch?v=3IqtilutjuU>  
[^95]: <https://www.youtube.com/watch?v=miCamquxhe4>  
[^96]: <https://www.joinsecret.com/google-ai-studio/alternatives>  
[^97]: <https://www.uptech.team/blog/how-to-build-ai-software>  
[^98]: <https://venturebeat.com/ai/google-gemini-ai-just-shattered-the-rules-of-visual-pro>  
[^99]: [https://www.reddit.com/r/AskProgramming/comments/1dbnhae/own\\_ai\\_chatbot\\_for\\_local\\_](https://www.reddit.com/r/AskProgramming/comments/1dbnhae/own_ai_chatbot_for_local_)  
[^100]: [https://www.reddit.com/r/node/comments/xbrtse/how\\_are\\_messages\\_stored\\_in\\_a\\_chat\\_a](https://www.reddit.com/r/node/comments/xbrtse/how_are_messages_stored_in_a_chat_a)  
[^101]: [https://www.reddit.com/r/webdev/comments/1gypy8o/what\\_is\\_the\\_correct\\_way\\_to\\_store](https://www.reddit.com/r/webdev/comments/1gypy8o/what_is_the_correct_way_to_store)  
[^102]: [https://www.reddit.com/r/LocalLLM/comments/1j3uhli/feedback\\_on\\_my\\_locally\\_hosted\\_](https://www.reddit.com/r/LocalLLM/comments/1j3uhli/feedback_on_my_locally_hosted_)  
[^103]: [https://www.reddit.com/r/selfhosted/comments/1fjoosq/self\\_hosted\\_backups\\_of\\_whats](https://www.reddit.com/r/selfhosted/comments/1fjoosq/self_hosted_backups_of_whats)  
[^104]: [https://www.reddit.com/r/LangChain/comments/1dj1che/newb\\_questions\\_maintaining\\_fu](https://www.reddit.com/r/LangChain/comments/1dj1che/newb_questions_maintaining_fu)  
[^105]: [https://www.reddit.com/r/selfhosted/comments/1c5mydq/an\\_overview\\_of\\_100\\_opensourc](https://www.reddit.com/r/selfhosted/comments/1c5mydq/an_overview_of_100_opensourc)  
[^106]: [https://www.reddit.com/r/ollama/comments/1eafc9h/i\\_built\\_a\\_stupidsimple\\_app\\_to\\_ch](https://www.reddit.com/r/ollama/comments/1eafc9h/i_built_a_stupidsimple_app_to_ch)  
[^107]: [https://www.reddit.com/r/AI\\_Agents/comments/1i2wbp3/whats\\_the\\_best\\_way\\_to\\_handle\\_](https://www.reddit.com/r/AI_Agents/comments/1i2wbp3/whats_the_best_way_to_handle_)  
[^108]: [https://www.reddit.com/r/cloudstorage/comments/1b0kfx8/what\\_cloud\\_storage\\_would\\_y](https://www.reddit.com/r/cloudstorage/comments/1b0kfx8/what_cloud_storage_would_y)  
[^109]: [https://www.reddit.com/r/softwarearchitecture/comments/16bk4w/software\\_architectu](https://www.reddit.com/r/softwarearchitecture/comments/16bk4w/software_architectu)  
[^110]: [https://www.reddit.com/r/AI\\_Agents/comments/1j2hlya/handling\\_chat\\_history\\_in\\_full](https://www.reddit.com/r/AI_Agents/comments/1j2hlya/handling_chat_history_in_full)  
[^111]: [https://www.reddit.com/r/LocalLLaMA/comments/1hmfd85/we\\_built\\_an\\_os\\_to\\_protect\\_ai](https://www.reddit.com/r/LocalLLaMA/comments/1hmfd85/we_built_an_os_to_protect_ai)  
[^112]: [https://www.reddit.com/r/lovable/comments/1jgmrw1/chat\\_functionality\\_and\\_no\\_local](https://www.reddit.com/r/lovable/comments/1jgmrw1/chat_functionality_and_no_local)  
[^113]: <https://stackoverflow.com/questions/49242625/chat-application-how-to-store-chat-h>  
[^114]: <https://community.openai.com/t/best-practices-for-chat-conversation-storage-and-c>  
[^115]: <https://www.pcmag.com/picks/the-best-cloud-storage-and-file-sharing-services>  
[^116]: <https://www.youtube.com/watch?v=etrN89C0yg8>  
[^117]: <https://optiblack.com/insights/ai-chatbot-session-management-best-practices>  
[^118]: <https://learn.microsoft.com/en-us/azure/azure-web-pubsub/tutorial-build-chat>

[^119]: <https://lumenalta.com/insights/12-best-practices-for-using-chatgpt-effectively>  
[^120]: <https://chatboxai.app/en>  
[^121]: <https://cloud.google.com/run/docs/tutorials/websockets>  
[^122]: <https://www.pinecone.io/learn/series/langchain/langchain-conversational-memory/>  
[^123]: <https://zapier.com/blog/best-cloud-storage-apps/>  
[^124]: [https://www.youtube.com/watch?v=SP-Y\\_90EaFg](https://www.youtube.com/watch?v=SP-Y_90EaFg)  
[^125]: <https://www.semanticscholar.org/paper/e3464eb2ab92449a39e7b6435d6aa056094a0f22>  
[^126]: <https://www.semanticscholar.org/paper/2c4ac009e09208fb5f9bd24c0f6a8ac482c7b097>  
[^127]: <https://www.semanticscholar.org/paper/7276c6efb32f5ebd18b9746e511f12e52f079b1c>  
[^128]: <https://www.semanticscholar.org/paper/b5f8e94450d21095d52d05ac6b8ab1deb1f8e9c4>  
[^129]: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10679402/>  
[^130]: <https://www.semanticscholar.org/paper/bca8e1d4314fd080f4b7c93529ce09d2543b8f34>  
[^131]: [https://www.reddit.com/r/GoogleGeminiAI/comments/1etxxcm/questions\\_about\\_integrat](https://www.reddit.com/r/GoogleGeminiAI/comments/1etxxcm/questions_about_integrat)  
[^132]: [https://www.reddit.com/r/Bard/comments/1hrre1v/can\\_somebody\\_explain\\_google\\_ai\\_stu](https://www.reddit.com/r/Bard/comments/1hrre1v/can_somebody_explain_google_ai_stu)  
[^133]: [https://www.reddit.com/r/Bard/comments/1jq3cyg/ai\\_studio\\_get\\_a\\_new\\_ui\\_redesigned/](https://www.reddit.com/r/Bard/comments/1jq3cyg/ai_studio_get_a_new_ui_redesigned/)  
[^134]: [https://www.reddit.com/r/Bard/comments/1ina1n4/is\\_it\\_possible\\_to\\_use\\_search\\_or\\_gc](https://www.reddit.com/r/Bard/comments/1ina1n4/is_it_possible_to_use_search_or_gc)  
[^135]: <https://www.instituteofaistudies.com/insights/how-to-use-google-ai-studio-for-beg>  
[^136]: <https://developers.google.com/learn/pathways/solution-ai-gemini-101>  
[^137]: [https://www.reddit.com/r/storage/comments/1axunz0/centralized\\_storage\\_system\\_for\\_](https://www.reddit.com/r/storage/comments/1axunz0/centralized_storage_system_for_)  
[^138]: [https://www.reddit.com/r/LocalLLaMA/comments/1broa8h/is\\_there\\_a\\_way\\_for\\_me\\_to\\_use](https://www.reddit.com/r/LocalLLaMA/comments/1broa8h/is_there_a_way_for_me_to_use)  
[^139]: [https://www.reddit.com/r/LocalLLaMA/comments/16y95hk/a\\_starter\\_guide\\_for\\_playing\\_](https://www.reddit.com/r/LocalLLaMA/comments/16y95hk/a_starter_guide_for_playing_)  
[^140]: [https://www.reddit.com/r/selfhosted/comments/1hudru7/databridge\\_opensource\\_local\\_](https://www.reddit.com/r/selfhosted/comments/1hudru7/databridge_opensource_local_)  
[^141]: <https://bluexp.netapp.com/blog/object-storage-block-and-shared-file-storage-in-gc>  
[^142]: <https://www.youtube.com/watch?v=hKrl5Gr7hM0>  
[^143]: <https://www.youtube.com/watch?v=A4daQj9tnWk>  
[^144]: <https://www.newhorizons.com/resources/blog/multi-cloud-storage>  
[^145]: <https://firebase.google.com/docs/vertex-ai/solutions/cloud-storage>  
[^146]: <https://www.semanticscholar.org/paper/e453d9acac6f616d24111929d550598904ac9da5>  
[^147]: [https://www.reddit.com/r/webdev/comments/1h095wr/i\\_dont\\_understand\\_how\\_they\\_build](https://www.reddit.com/r/webdev/comments/1h095wr/i_dont_understand_how_they_build)  
[^148]: [https://www.reddit.com/r/LocalLLaMA/comments/1847qt6/llm\\_webui\\_recommendations/](https://www.reddit.com/r/LocalLLaMA/comments/1847qt6/llm_webui_recommendations/)  
[^149]: [https://www.reddit.com/r/ChatGPTCoding/comments/1i3265w/best\\_ai\\_developer\\_tools\\_v](https://www.reddit.com/r/ChatGPTCoding/comments/1i3265w/best_ai_developer_tools_v)  
[^150]: [https://www.reddit.com/r/GoogleGeminiAI/comments/1dvl059/gemini\\_ai\\_studio\\_is\\_brok](https://www.reddit.com/r/GoogleGeminiAI/comments/1dvl059/gemini_ai_studio_is_brok)  
[^151]: <https://www.uxdesigninstitute.com/blog/the-top-8-ai-tools-for-ux/>  
[^152]: <https://codeparrot.ai/blogs/best-design-to-code-ai-tools-for-developers>  
[^153]: <https://www.linkedin.com/pulse/transforming-figma-ui-designs-jetpack-compose-code>  
[^154]: <https://www.pragmaticcoders.com/resources/ai-developer-tools>  
[^155]: [https://www.reddit.com/r/FlutterDev/comments/1cuobwb/local\\_storage\\_architecture/](https://www.reddit.com/r/FlutterDev/comments/1cuobwb/local_storage_architecture/)  
[^156]: [https://www.reddit.com/r/LangChain/comments/1c0xi7a/advice\\_needed\\_on\\_orchestrati](https://www.reddit.com/r/LangChain/comments/1c0xi7a/advice_needed_on_orchestrati)  
[^157]: <https://www.youtube.com/watch?v=kpjwQDW0TFg>  
[^158]: [https://docs.llamaindex.ai/en/stable/module\\_guides/storing/chat\\_stores/](https://docs.llamaindex.ai/en/stable/module_guides/storing/chat_stores/)  
[^159]: <https://www.vellum.ai/blog/how-should-i-manage-memory-for-my-llm-chatbot>  
[^160]: <https://cloudian.com/ai-workflows/>  
[^161]: <https://docs.typingmind.com/cloud-sync-and-backup/cloud-storage>