□ AI Usage Report – ADmyBRAND Dashboard

□ Overview

The **ADmyBRAND Dashboard** integrates an intelligent assistant that simulates an AI-powered data analyst, designed to assist users with interpreting campaign metrics, trends, and performance insights.

Although the current version uses **mock responses**, the architecture is built to seamlessly support real-time AI responses using services like **Google Gemini API**, **OpenAI GPT-4**, or **Anthropic Claude** in the future.

□ AI Tools & Libraries Used

Tool/Library	Purpose
`React`	Frontend framework for component-driven UI
`axios`	HTTP client for API calls (planned use for real AI integration)
`setTimeout()`	Simulates network delay for mock AI response latency
`useState`, `useEffect`	React hooks to manage AI conversation state and loading indicator
`Tailwind CSS`	Design system for styling AI response bubbles and chat layout
`lucide-react`	Used to render bot and metric icons with semantic styling

---

□ Mock AI Assistant

□ Current Behavior

The mock AI assistant provides pre-defined responses randomly selected from an array. This enables:

- Fast prototyping
- Visualizing the chat flow
- Testing loading spinners and UI conditions
- Allowing developers to extend easily to real APIs later

□ How it Works

```
const mockResponses = [
  "Your campaign performance shows a 12% increase in conversions compared to last week.",
  "The seasonal campaign is performing exceptionally well with a ROAS of 3.2x.",
  "Consider increasing budget for the top performing campaigns.",
  "Your bounce rate has decreased by 5% since last month - great job!",
  "I recommend focusing on mobile users as they show higher engagement rates."
];
```

- The AI chat function (`sendMessage`) stores the user's message, simulates loading (`setIsLoadingAI(true)`), then randomly selects a mock response after a 1-second delay using `setTimeout`.

□ Planned AI Integration (Real APIs)

Option 1: **Google Gemini API**

Replace the mock logic with:

```
const response = await axios.post(
  'https://generativelanguage.googleapis.com/v1beta/models/gemini-pro:generateContent',
  {
    contents: [{ parts: [{ text: message }] }]
  },
  {
    headers: {
      'Content-Type': 'application/json',
      'Authorization': `Bearer ${process.env.VITE_GEMINI_API_KEY}`
    }
  }
);
const botMessage = response.data.candidates[0].content.parts[0].text;
```

Option 2: **OpenAI (GPT-4)**

```
const response = await axios.post(
  'https://api.openai.com/v1/chat/completions',
  {
    model: 'gpt-4',
    messages: [{ role: 'user', content: message }]
  },
  {
    headers: {
      'Authorization': `Bearer ${process.env.OPENAI_API_KEY}`
    }
  }
);
const botMessage = response.data.choices[0].message.content;
```

□ Suggested Prompts for Real AI

- "Give me a summary of this week's top performing campaigns."
- "Why has my conversion rate dropped on mobile devices?"
- "Suggest optimizations based on the dashboard trends."
- "What's the ROAS trend over the last month?"
- "Which channel has the highest cost per acquisition?"

These prompts can be extended using NLP for structured queries.

□ Conclusion

The ADmyBRAND Dashboard lays a strong foundation for AI integration. It uses a clean modular approach where mock logic can be replaced by real API endpoints. This design ensures that minimal refactoring is required to upgrade the assistant from static responses to dynamic, LLM-powered insights.