ChatGPT

Recommended MVP Domains for an AI Decision-Tree Assistant

After evaluating several domains, we recommend the following for an initial MVP (ranked by priority):

1. **Tech Support / Troubleshooting (Consumer)** – Guiding non-technical users through device or software issues (e.g. "Why won't my laptop connect to Wi-Fi?"). Decision-tree dialogs are already common in support chatbots because they handle simple yes/no branches effectively [1]. A dynamic tree can ask the right questions (e.g. device type, symptom details) and lead the user to a solution (power check, settings, etc.). This delivers clear value: users get quick fixes without waiting for human support. Non-linear decision trees are explicitly noted as ideal for troubleshooting technical problems with multiple causes [2]. Even simple flows (power on/off, cable checks, basic resets) cover many frequent issues.

2. **Education (Course/Learning Path Advice)** – Helping learners or students choose courses or career paths based on interests, prerequisites, and goals. For example, a user might ask, "I enjoy math and want a job in AI – what should I study next?" The assistant could ask about current level, topics of interest, and suggest relevant courses or learning plans. This is consumer-friendly and structured: many students struggle with course planning, and chatbots for academic advising (e.g. course recommendations tailored to a student's program and progress) have proven useful [3]. Even a narrow focus (e.g. "which programming class to take") provides immediate value.

3. **Personal Finance (Budgeting / Savings Advice)** – Guiding users through simple money-management decisions. For instance, answering "How should I adjust my budget if I cut expenses by 10%?" or "Should I pay off debt or save?" could be structured as a decision tree. AI chatbots (like Cleo) have shown that people want to ask natural-language finance questions (about spending, saving, investing) and get concrete answers [4]. Starting with a narrow scope (e.g. monthly budgeting or basic savings plan) keeps complexity low. This is consumer-friendly (many enjoy money tips) and business-agnostic (it applies to anyone's personal budget).

4. **Health & Wellness (Fitness Plans or Symptom Triage)** – Offering guided advice on minor health/wellness issues. Examples include a fitness-plan chooser ("I want to lose weight – what exercise plan fits me?") or a simple symptom checker ("I have a sore throat and fever – should I rest at home or see a doctor?"). Chatbots have long used decision-tree Q&A for health triage (e.g. Ada Health walks users through a structured questionnaire about symptoms) [5]. A narrow MVP could focus on wellness or general fitness (avoiding medical complexity) – e.g. a workout-suggestion flow based on user goals, time, and equipment. Meal-planning is another safe subdomain: Lark's AI coach uses conversational flows to help members plan meals tailored to diet and preferences [6]. Users get clear value (immediate health or diet tips), and the structured nature of health decisions maps well to branching questions.

5. **Lifestyle (Food or Outfit Recommendations)** – Helping users make everyday choices through interactive Q&A. For example, a “what should I eat for dinner?” assistant could ask about hunger level, dietary restrictions, cuisine preferences, etc. (as illustrated by a simple decision-tree food-choice example [7]). Or a “what to wear” assistant might ask about occasion and weather. These flows are straightforward to build and very consumer-friendly. The value is modest but clear (meal ideas or outfit guidance), and even a very limited scope (e.g. deciding between a few options) can demonstrate the concept. Such use cases are often the simplest branching dialogs to implement (food-choice trees are literally used as examples of decision-tree chatbots [7]).

Each of the above domains involves structured decisions and provides immediate user benefit in a limited context. In particular, **Tech Support/Troubleshooting** emerges as the top pick for an MVP, because its flows are well-defined, common questions are predictable, and even a simple diagnostic assistant can resolve many issues. We now detail a specific use case in that domain.

#1 Domain – Tech Support / Troubleshooting

- **Specific Use Case:** A virtual assistant for common consumer device problems. For example, helping a user diagnose and fix a home Wi-Fi issue, computer connectivity, or smartphone problem. The user starts with a free-text question like “My computer is not turning on,” and the assistant asks follow-up questions to narrow the cause (e.g. “Is it plugged in?”, “Any indicator lights?”, “Have you tried holding the power button longer?”). It then recommends next steps (check power, reset modem, contact support, etc.). Focusing on a narrow set of everyday problems (power issues, internet setup, printer jams, etc.) keeps the scope low while demonstrating utility.

- **Why It’s Ideal:** Technical troubleshooting naturally follows a decision-tree pattern, making it easy to model step-by-step logic. Branching logic is already common for these tasks: for instance, chatbots often handle FAQ-style tech support by “yes/no” questions to guide users to a solution [1]. The user value is immediate: people want quick fixes without deep technical knowledge. Even a simple assistant can resolve a high percentage of routine queries (power-cycle steps, configuration checks) and reduce support calls. As one guide notes, non-linear decision trees are *“ideal for scenarios requiring diverse outcomes”* like troubleshooting tech issues [2]. By using AI to generate and refine the tree on the fly, the MVP can cover more scenarios than a fixed script would, without needing to pre-code every path. Building this is relatively easy: start with a few device categories and common symptoms.

- **User Personas:** Typical users might include *everyday consumers and home users* who lack technical expertise. For example, a middle-aged parent whose home router isn’t working, an office worker with a malfunctioning printer, or an elderly person whose TV won’t turn on. All are non-technical “power users” who want step-by-step guidance. (A persona might be “Tom, 45, non-IT small business owner who needs to fix his slow PC.”) These users prefer a guided chat over reading manuals or waiting for a technician. The assistant should use simple language and avoid jargon – an area where AI explanation can help.

- **Sample Decision Flow:**

- **Step 1:** Identify device and symptom. *“Which device is having an issue? (e.g. laptop, router, printer)”*

- **Step 2:** Clarify problem category. *“Is the device not turning on, not connecting to internet, printing blank pages, etc.?”*
- **Step 3:** Ask contextual questions. E.g. if “not turning on”, ask about power lights or battery status; if “no internet”, ask if other devices have connectivity.
- **Step 4:** Offer solution suggestions. Based on answers, the flow branches into concrete actions: “Please plug the charger in firmly and try again,” or “Restart your router by unplugging it for 10 seconds,” etc.
- **Step 5:** Confirm outcome or escalate. If the fix works, end the session; otherwise offer alternate steps or “connect to human support.”

This flow can be presented as a hierarchical tree, where each answer leads to the next question or solution. For example, a flow might start: “Device = Laptop” → “Symptom = Won’t power on” → “Check if power cable is connected” (yes/no) → branch to “Try another outlet” or “Proceed to battery check.”

- **Where AI Helps:** AI adds value in *dynamically generating and adapting the tree*, and in *clear explanations*. Unlike rigid scripts, an AI assistant can interpret the user’s free-text replies and steer the flow flexibly. It can rephrase follow-up questions based on context, handle unexpected answers, and clarify technical steps in plain language. For example, if a user says “The wifi light is off,” the AI can ask, “It sounds like your router’s internet light is off. Have you tried rebooting the router?” – even if that exact question wasn’t pre-coded. AI can also refine questions on the fly (“Do you see any error message on screen?”) and summarize solutions at each step. In short, AI is used to **build the decision tree in real time** and to **explain/translate technical details**, while the basic branching logic provides structure. This hybrid approach (structured decision points + AI-language) aligns with best practices for chatbots [2] [1] and allows the MVP to handle a wider range of issues than a static flow would.

Sources: We looked at examples of chatbot decision-tree use in various domains. For instance, tech support flows often use decision-tree Q&A for troubleshooting [2], and AI chatbots in education already offer course recommendations through similar branching logic [3]. Consumer-facing finance bots let users ask natural questions about budgets and get helpful answers [4]. In health apps, Ada Health exemplifies a symptom-checking decision tree [5], and wellness coaches use AI flows to plan meals or workouts [6]. These cases show that a focused MVP can be both simple to build and immediately useful.

---

[1] Diagnostic vs. Decision Tree Approach: Which is Better for Support?
https://www.mavenoid.com/en/blog/diagnostic-vs-decision-tree-approach-which-is-better-for-support

[2] Chatbot Decision Tree: Types, What to Consider & Tips
https://livechatai.com/blog/chatbot-decision-tree

[3] The Role of AI Chatbots For Higher Education Success In 2024 | Element451
https://element451.com/blog/chatbots-in-higher-ed-what-you-should-know

[4] ChatGPT-based apps like Cleo give surprisingly sounds financial advice | Vox
https://www.vox.com/tech-policy/421783/chatgpt-financial-advice-ynab-monarch-cleo

[5] Eureka Health vs Ada Health: Which AI doctor and symptom checker app should you pick?
https://www.eurekahealth.com/resources/eureka-health-vs-ada-health-which-ai-doctor-and-symptom-checker-app-should-you-pick-en

6 Lark’s Coach+ and the Evolution of Food Logging & Meal Planning - Lark Health Blog
https://www.lark.com/resources/larks-coach-and-the-evolution-of-food-logging-meal-planning

7 Propel PRM
https://www.propelmypr.com/blog/ai-lets-demystify-the-buzz