Tech Innovators - README

1. Problem Statement

Current AI chatbots (ChatGPT, Gemini) provide outdated/static information. Users still need to manually open browsers, search, filter, and copy data. This wastes time and effort. We need an AI Agent that reduces these steps and delivers fresh, real-time answers.

2. Detailed Proposal and Prototype Plan & Features

We propose an AI Agent that understands natural language instructions, uses a local LLM for planning, automates the browser to fetch live information, and returns results in structured formats (tables/lists).

1 Understands natural language instructions
2 Local LLM for task planning
3 Browser automation to fetch live data
4 Structured output formats (tables, lists, links)
5 Example: 'Search laptops under ■50,000' → Agent browses and returns top 5 results with prices

3. Tech Stack

1 Orchestration: Python / Node.js
2 Instruction Parsing: LangChain + Ollama (local LLM)
3 Browser Automation: Playwright / Selenium / Puppeteer
4 Output: JSON → Table/Chat UI

4. Team Contributions

1 M. Siva Harsha Vardhan – Prepared the PPT presentation
2 G. Chandu – Created innovative solutions to the problem
3 N. Ravi Kumar – Debugged errors and rectified issues
4 K. Yuva Rani – Worked on web automation
5 P. Anjali – Developed the complete code using Python
6 ChatGPT – Assisted in documentation and report preparation

5. Impact & Use Cases

The solution saves users time by eliminating repetitive browsing tasks. It provides fresh, structured, real-time information for multiple domains.

1 Students – Research laptops, books, and courses
2 Shoppers – Product comparisons
3 Professionals – Fetch latest stock prices and news
4 General users – Save time and avoid manual browsing

6. Future Scope

1 Voice-based commands
2 Multi-language support

3 Deeper integrations (forms, emails, job portals, etc.)

■ *Thank You*