May 10, 2026

AskBeevs - the Return

Summary

The session showcased AI-assisted application development, exploring coding workflows and organizational strategies for leveraging agentic tools.

Application Development Workflow
Participants used Claude Code to design and build an interview preparation application. The process involved defining product requirements to generate architecture and leveraging terminal interfaces for deployment.

AI Debugging and Capabilities
The discussion highlighted the iterative power of LLMs to self-correct code and address functional errors. The group confirmed that agents can effectively manage recursive tasks and handle multilingual translations.

Strategic AI Organizational Adoption
The primary decision reached was that organizations should prioritize empowering staff with AI tools over creating restrictive usage barriers. Investment in individual AI productivity remains essential for future competitiveness.

Next steps

- [ ] [Timothy Beevor] Delete Key: Delete the API key produced during the demonstration immediately after the meeting concludes.

Details

- **Building an AI Application for Job Interview Preparation**: Timothy Beevor introduced the session's objective, which was to build an application using AI to help people prepare for job interviews. They explained that the approach is simple and can be applied to any problem, emphasizing that people often do not try this process because they assume they lack the necessary skills. The core idea is to use AI, whose language is code, to brainstorm and develop the application, thereby allowing people to achieve utility they may not have had before.
- **Framing the Application Idea and Design Generation**: The speaker framed the product they wanted to build as an application to help students practice for job interviews, allowing configuration of student information (resume, skills, goals) and job details (position, company, level). The application was also envisioned to allow configuration of the interviewer's personality (e.g., tough, easy) and simulate a chat with the Anthropic key. This detailed description was provided to the AI with a request for a sensible design, user interface, and architecture, which the AI then generated, including a plan, data models, file structure, and build order.
- **Setting Up the Coding Environment using Claude Code**: Timothy Beevor demonstrated how to initiate the build process by using the command prompt interface for Claude Code, which is accessed via the terminal and pointed to a project folder. The speaker explained that they prefer using the command prompt, although Claude Code can also be used through the code tab in the Claude app. They issued a prompt to Claude Code, asking them to use the Anthropic Claude API to build the application according to the provided specification file.
- **API Key Management and Security Considerations**: During the application setup, Bruce inquired about the Claude Code API key, which Timothy Beevor explained is a metered gate for accessing the coding engine, and although very cheap for simple uses, the key must be protected. The speaker noted that they limit their token spending so the service automatically turns off when the

tokens run out, which is necessary because if a key is compromised, someone could exploit it to incur high costs.

- **Comparison of AI Code Tools and Utility**: Bruce asked how Claude compares to Copilot, to which Timothy Beevor responded that Claude currently "annihilates it," although Microsoft is moving in a similar direction. The speaker also mentioned OpenAI's Codeex and Google's anti-gravity as equivalent tools. They also noted that users can set the effort level for Claude Code to "very high" to encourage them to make fewer mistakes.
- **Exploring Alternative Deployment and Data Handling Options**: The application continued coding, and Timothy Beevor demonstrated how the same process could be done using Replit.com for easier deployment and server management, noting that this option costs approximately $15-$20 a month. The speaker clarified that while the current app likely does not have a database and saves data only for the session, a user could instruct Claude Code to add a user account and database for persistent data storage.
- **AI Agent Capabilities in Language and Interface Design**: Sharron Jackson asked if Claude supports other languages, and Timothy Beevor confirmed that the language models are very good at translation and can be used to conduct tasks, such as interviews, in languages like French. Timothy Beevor also explained that the AI can build custom front-end components and is capable of interacting with APIs, allowing them to perform tasks like pulling and analyzing data from social media in a different language to write reports in English.
- **Testing and Debugging the Application**: Timothy Beevor demonstrated the initial execution of the application, which required manually setting the API key as guided by Claude Code. They noted that the AI retains a memory bank of the user's experience (e.g., being a newbie) for future interactions. When the application encountered a bug during the profile saving process, the speaker explained that any errors could be fed back to the agent for debugging and fixing.
- **Debugging a Software Application and the Power of LLMs**: Timothy Beevor, Sharron Jackson, and Bruce discussed an issue where an application may have written all the content but lacked a button to enter an interview, suggesting a potential forgetting to put a crucial element somewhere. Sharron Jackson observed the ability of large language models (LLMs) to fix their own errors when building an application in Replit, which Timothy Beevor identified as a key strength of AI, particularly their capability for recursive work in coding by breaking down a problem into steps and checking each one. Timothy Beevor suggested that the real power of AI lies not just in answering

questions but in building new products and services that can change global operations, and they also suggested that AI companies might be cooperating more than commonly perceived.

- **AI Adoption and Organizational Strategy**: Timothy Beevor opined that organizations should shift their focus from building barriers around AI ethics and committee-driven use cases to empowering every staff member with skills that could make them orders of magnitude more productive. They suggested that the "north star" for many organizations should be helping humans solve problems better, rather than treating AI integration strictly as a massive IT project requiring full certification. The speaker cited examples in education where simple AI-enabled tools helped teachers with curriculum adaptation, note-taking conversion to emails, and building interactive agents for teaching concepts like parabolas, demonstrating the simple, impactful changes AI can bring.
- **Investment and Normalization of AI Tools**: Timothy Beevor emphasized the endless creative potential of AI agents, which can handle tasks like voiceovers, video creation, and building presentations, and encouraged people to start playing with basic tools to normalize the technology before moving to sensitive data. They noted that many basic tools and APIs, such as the command line, are free, and even commercial tools are inexpensive, suggesting organizations should view AI as a human resources equation. The speaker proposed that investing around \$2,000 per employee annually for AI tools, equating to a small pay increase, would significantly boost productivity, arguing that the future will require all organizations to change to remain competitive and efficient.
- **Application of AI in Various Industries and Debugging Code**: Timothy Beevor asserted that AI is useful across all industries, including finance, HR, and farming, and discussed how users could build agents to analyze markets, crop data, and pricing. They noted that a key difference in using AI for coding is the ability to easily ask the AI for help when encountering errors, allowing the AI to recursively fix the code or write a prompt for an agent to fix issues caused by context window limitations. Timothy Beevor highlighted the immense value of continuous iteration and feedback with AI, noting that the agent never gets frustrated, which ultimately helps users achieve their goals.
- **Demonstration of Full Self-Driving Technology**: Timothy Beevor shared an experience using full self-driving technology in a Tesla, describing the car's ability to smoothly navigate challenging mountain highways, use regen braking precisely, and accelerate flawlessly. They noted that the experience felt safer than human driving and provided the driver with significantly more time, although they confirmed that the system is programmed to prompt the

human driver to take over if it encounters uncertainty. Bruce agreed that the technology felt good after they had briefly taken their hands off the steering wheel.

- **Successful Creation of an App Using Replit**: Sharron Jackson confirmed that they successfully created an application using Replit during the meeting, noting that while it still had errors, the rapid development was impressive. The application they built was designed to generate sewing patterns and instructions based on a person's measurements and an uploaded picture of the desired item. Sharron Jackson reported the app generated instructions and a download button for a PDF, and the agent was actively fixing a missing download button for the pattern pieces, noting that the code for a minimum viable product was written in about eight minutes.

You should review Gemini's notes to make sure they're accurate. Get tips and learn how Gemini takes notes

*How is the quality of **these specific notes?** Take a short survey to let us know your feedback, including how helpful the notes were for your needs.*