NITIC News Article
Blog Post Published 09.18.2025
https://www.nitic.org/vibe-coding-will-it-change-how-we-teach-programming/

Vibe Coding – Will it Change How We Teach Programming?

Rajiv Malkan, Ph.D.; Professor, Data Analytics, AI and Computer Science; Co-PI, NSF Funded, National Information Technology and Innovation Center (NITIC); Lone Star College – Montgomery, 3200 College Park Drive; rajiv.malkan@lonestar.edu

Vibe Coding is an emerging paradigm in software development that leverages large language models (LLMs) and other AI assistants to generate functional code from natural language prompts. Open AI Co-Founder and Ex-Tesla AI Senior Director Andrej Karpathy coined the word 'Vibe Coding'. He wrote on X, developers can 'give in to the vibes, embrace exponentials and forget that the code even exists."

Vibe Coding, describes a workflow where the primary role shifts from writing code line-by-line to guiding an AI assistant to generate, refine, and debug an application through a more conversational process. This frees up the programmer to think about the big picture, or the main goal of the application development, while the AI handles writing the actual code.

According to the survey done by YC Startups Founders, 95% of the new code is now generated by Large Language Models (LLM), in other words – Natural language prompts are generating most of the code. According to the survey in past month, the speed of software development has gone up 100X due to implementation of Vibe Coding concepts at most of the startups. Check out this video – https://www.ycombinator.com/library/ME-vibe-coding-is-the-future.

The recent advent of specially created low code/no code – AI coding assistants, like GitHub Copilot, Google AI Studio, Firebase Studio – catalyzed the shift towards Vibe Coding. These assistants can make real-time predictions about what you're trying to do and offer intuitive suggestions to make it easier than ever to create software, even if you've never written code before. "Over the past three or four years, these AI autocomplete tools have become better and better—they started off completing single lines of code and can now rewrite an entire file for you,

or create new components," says Barron Webster, a software designer at the interface company Sandbar. "The remit of what you can take your hands off the wheel and let the machine do is continually growing over time."

How Vide Coding Works:

Typical workflow to develop a simple code using vibe coding concepts:

1. Describe the goal: Programmer will start with a high-level prompt in plain language. For example: "Create a Python function that reads a CSV file."
2. AI generates code: The AI assistant interprets the programmer's request and produces the initial code.
3. Execute and observe: Programmer runs the generated code to see if it works as intended.
4. Provide feedback and refine: If the output isn't quite right or an error occurs, programmer provides new instructions, like, "That works, but add error handling for when the file is not found."
5. Repeat: This loop of describing, generating, testing, and refining continues until the code is complete.

As one can see from the above AI code generating workflow process, the programmer has not written a single line of code. Through conversational prompt dialogues the software application is developed.

As an educator, we need to ask, how does it impact teaching coding and programming in the future? In the next blog we will dig deeper into Vide Coding concepts by discussing how to use AI assisted tools to generate code.

References:

https://www.ibm.com/think/news/vibe-coding-devs
https://www.ycombinator.com/library/ME-vibe-coding-is-the-future
https://cloud.google.com/discover/what-is-vibe-coding
https://www.technologyreview.com/2025/04/16/1115135/what-is-vibe-coding-exactly
https://build.microsoft.com/en-US/sessions/BRK102
Using GitHub Copilot in your IDE: Tips, tricks, and best practices – The GitHub Blog

---

This material Is based upon work supported by the U.S. National Science Foundation under Grant No. 2300188. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.