**Assignment Activity Unit 3**

Kelechi Nwa-uwa

UNIV 1001-02

Wednesday, September 24, 2025

### PART 1

Technology is changing everything in the world. From the ways we communicate, to our kitchen appliances, to the tools used in education. It is no surprise that these changes are also present in the career making some of this revolutionary technology possible– Software Engineering (SE). The recent breakthroughs in Natural Language Processing (NLP) and Large Language Models (LLMs) especially have introduced significant transformations in how SE works.

Kumar and Chimalakonda (2024) propose “a FedLLM for ‘code summarization’ as developers spend more time in comprehending code.” This model cuts down the time spent trying to understand code, making it vital if it achieves wide adoption. Shorter comprehension time for developers means that code gets shipped much faster; bug fixes and feature releases are rolled out quicker.

While this is still a proposition, the current versions of LLMs can generate syntactically accurate code, and even planning system architecture. Chat purpose models like ChatGPT are capable of this, and there are models made specifically for code generation. These models are the basis for software like LetsCommit which aids the documentation of code changes, saving the developer valuable time (Neyem et al., 2024).

The changes these LLMs bring are not all positive. Artificial Intelligence (AI) has its limitations and these can be very costly when the tools are used to aid SE. With AI-assisted SE gaining wider adoption, there are concerns about privacy; higher likelihood of technical debt; and a precarious job market that seeks only senior developers with prompt engineering experience.

LLMs are shaping the way developers think about code, architect code, and even document code changes. This naturally has its positives and its negatives. Technology is advancing at a rapid rate, transforming human lives and redefining our collective baseline. It begs the question, are other human infrastructure advancing enough to keep up with our technology?

### PART 2

Giving credit to the authors of the articles is important because it demonstrates academic integrity. The offense of plagiarism exists because every author deserves acknowledgement for their original ideas.

Representing the words, thoughts or ideas of another author as one’s own is plagiarism. This is the most common type of academic dishonesty. Another common type of academic integrity is fabrication, which involves representing facts, data, citation, or documents in a way that is misleading. Unauthorized assistance involves academic work being carried out partially or fully by a third party; misrepresentation involves fraud for the purpose of academic gain; and collusion is aiding in any type of academic dishonesty. (University of the People [UoPeople], n.d.)

I have used Artificial Intelligence (AI) to draft an outline for my essay in Part 1. I have also used AI to check my grammar and spelling, and to ensure that my citations follow the correct APA format. In order to ensure academic integrity, I kept my prompts confined to technical adjustments and planning. This reduces the likelihood of me including unverified information in my work. I also cross checked its suggestions with the APA citation style guide on the Learning Resource Center, to avoid unintentional fabrication or plagiarism. (Georgetown University Library, 2025)

### 

### PART 3

I have found the instructor’s feedback quite helpful. It helps me understand my instructor’s expectations better, and how to meet them. While the grade was high, I am aware that there is always room for improvement. I am grateful for the feedback, which pointed out technical weaknesses such as the correct document formatting; as well as my sparse use of citations.

I believe that by implementing the suggestions in these feedback comments, I will improve my academic writing skills and gain more knowledge.

**References**

Georgetown University Library. (2025, January 27). *Artificial Intelligence (generative) resources*. <https://guides.library.georgetown.edu/ai/ethics>

Kumar, J., & Chimalakonda, S. (2024). Code summarization without direct access to code - Towards exploring federated LLMs for software engineering. In *Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering (EASE ’24)* (pp. 100–109). Association for Computing Machinery. <https://doi.org/10.1145/3661167.3661210>

Neyem, A., Ríos-Letelier, A., Céspedes-Arancibia, K., Sandoval Alcocer, J. P., & Mendoza, M. (2024). Enhancing commit message quality in software capstone projects with generative AI. *SoftwareX, 28*, 101947. <https://doi.org/10.1016/j.softx.2024.101947>

University of the People. (n.d.). *UoPeople catalog* . <https://catalog.uopeople.edu/>