**In your own words, what is the objective of technical writing, and why may it be useful in the context of a Cloud Computing and Cloud Security course? How might it benefit a business or a clients' users? Anticipated responses would be about 2-3 moderate sized paragraphs.**

The objective of technical writing is to provide complex information to readers in a manner they can understand and apply successfully without prior knowledge. In the context of a Cloud Computing and Cloud Security course, learning best practices for technical writing is equally important as the technical skills students learn and go on to apply in industry. In the modern workplace or any organized group of people working collaboratively to continually push towards stated goals, the lack of effective communication implies a range of time and monetary costs due to the resulting inefficiency. Students who learn technical skills successfully but fail to document their code and documentation in an easy to understand and reproducible way are lacking in their ability to collaborate with others effectively. Communicating effectively with colleagues orally, through code, and through documentation creates value and without it, that value is lost.

The benefits of technical writing scale from the individual up through the business in a way that is sustainable and adds to the institutional knowledge base. When an individual is clearly documenting why and what code is doing, it creates practical, replicable knowledge for anyone reading the code. The individual benefits from adhering to best practices for effective communication in a specified role include the ability to self-edit and easily understand anything individually created the ability to have peers review work for constructive feedback efficiently before deploying anything into production, and the ability to surmise documented work into presentation formats that is digestible for targeted internal or external audiences. The business benefits because the employee who is communicating effectively is adding clear and replicable technical knowledge to an ever growing and evolving body of institutional work that can be drawn from for many benefits. On the client-side, by communicating effectively with colleagues that represent the product or with clients directly, the technical-skilled employee can instill the resources and confidence in the end-user to use the product successfully.

**How might the documentation you write for your own development or operations team (system documentation) differ from the documentation you produce for end users (how to use/operate the system)? Anticipated responses would be about 2-3 moderate sized paragraphs and address the differences in audiences for documentation.**

The documentation I would write for my own personal development projects would differ from writing system documentation for an operations team or guides instructing end-users how to operate a system. The documentation that I write for my own personal projects is more verbose and informal by way of stream-of-thought comments on what and why I’m coding something rather than formal, clear prose I write for others to read. I tend to use all lower-caps for typing speed-efficiency since most of the time I am iterating a project for myself rather than finishing code for others to review by a specified deadline. I find it easier overall to be verbose with my comments limited to what a sub-unit of code is doing and why so that I can review everything at the end and edit out anything unnecessary that isn’t conducive to clear, understandable language and technical reproducibility.

The documentation I would write and have written for system documentation coincides with instructing end-users how to use, inspect, and operate revenue generating systems. Granted, these end-users were internal and limited to the information technology team of analysts, database administrators, data engineers, and managerial staff. The documentation I wrote for this audience and would write for a similar audience pertaining to a revenue generating system includes providing a bulleted summary of what a particular algorithm in the system does and why, a visual flowchart that embeds code and instructions at every node with a companion text file as an instructional guide, and clearly labeling the metadata associated with the file anywhere it lives in the system. Going forward, I want to apply adding a disclaimer of who the proceeding information is meant to be read by and implement the SOAR (situation, opportunity/obstacle, action, result) method to help improve my summarization of projects, any meaningful piece of code or shared document, and data storytelling presentation skills.

**In relation to the material and topics in this assignment, reflect on your own writing experience and capabilities, and consider "what is an area of my own writing, in a technical writing / documentation sense, that can be improved, and what topic from the material can help me accomplish this? Anticipated responses would objectively assess the student's own writing capabilities, identify an area that can be improved, and suggest how they can improve. A length of 1-3 paragraphs would be appropriate.**

I had over four years of information technology exposure as a business analyst for a higher education institution and although the workload was not as intense nor as collaborative as a prior business analyst internship for a Fortune top-fifty technology department, it was clear to me how important it was to effectively distill complex information directly with colleagues, managers, and orally in presentations or meetings. In my roles, I was primarily tasked with conducting meetings with varying department managers and gathering technical requirements for data or systems related requests. Listening to technical, or otherwise, jargon from managers and transforming it into an instructive, testable, technical document for our back-end engineers to work on is a key professional skill I have learned to this point. Moreover, socializing technical work documents with my team before a two-week sprint began and then again in a review meeting after the sprint for iterative feedback helped me improve my technical writing and communication skills drastically.

Now that I am formally developing programming skills and realizing the importance my documentation can have on colleagues, the business, end-users, and my own self growth, it is easy to see the continued importance on developing effective communication skills since it organizes people in a successful direction and is a key growth factor that develops leaders. Fortunately, to start my career I noticed the tremendous need for effective communication, channels for communicating effectively conducive with business goals, and clear business processes for efficient workflows. The management framework in both roles contrasted in how business processes, workflow, documentation, and chains of communication were instituted and enforced, and it has given me strong direction in how to hone my skills in this area going forward.

The reading attached to this assignment is a great reference to return to for tips on technical writing. As stated, I tend to be verbose with my thought formulation, so the point of limiting basic code explanations and elaborating on less-intuitive pieces of code struck me as something I should include in my personal style guide, especially as a best practice to carry over into industry. Providing snippets of how and how-not something is supposed to work is also a key point I think is particularly useful for instructing myself while upskilling my coding ability and for others to be able understand easily and replicate. Using task based headings with the audience range in mind is something that can help me be more concise with more effective, less verbose, word choice. And last, as I learn more programming and applications, not using anthropomorphism will help my writing style lean formal instead of conversational since much of my work will be collaborative and reviewed by colleagues or end-users. Having technical ability is great, but also being an effective technical communicator increases the ability to work with others, the ability to contribute to and grow knowledge bases, and personal growth above all.