

Computer Engineering: A Hard Task but Worthwhile Job

Computer Engineers and especially IT people do more than most people think, and in some cases do tedious amounts of work. During this assignment I interviewed my uncle, a senior software engineer, Ameer Ghobrial. I also got some resources from Michael Nickola (Management Level Engineer) and Vivek Ramesh (CEO of Slidenerd). However, for this assignment I will focus specifically on my uncle. People who do Software or Hardware engineering have to do: Digital Logic, Computer Design, Analog Automation, VHSIC Technologies, IC System Design, VLSI, Software modeling and UML Communication languages. They also have to learn Oracle or Cisco based systems if they want to work in both. Overall, depending on your CSE profession you will have much to learn and throughout this paper, I will explain why.

Now due to the confidentiality of the research and projects my uncle is pursuing in, I can't give you specific examples of writing in his field in terms of papers he did. Instead in this paper, I will be talking about the field in general, the topics in the field, and then in the rest of the paper conclude how it all ties in with the writing process and formal academic process in the Computer Engineering as well as Engineering field in general. This way I will let you be informed about the field's background, while still mentioning appropriate forms and writing styles in the field.

First of all I want to discuss, "What is Computer Engineering", as defined by my uncle: Computer Engineering is the combination of Hardware (Electrical) and Software (Computer Science and IT) to utilize or create new types of technologies. (Usually new hardware or types of

databases). Computer Engineering is a vastly growing field with many different jobs such as Senior Networking Engineers to Hardware Designers to Databases Managers or Developers. In fact, the job market is probably being flooded with more jobs than ever before. Every company right now either needs an app, or tech support, or to manage databases, etc. Web based and software companies in general such as Salesforce, also require Computer Engineers and IT majors to secure or do maintenance for their databases.

After interviewing my uncle, it made me start thinking “Why do I want to be a Computer Engineer or at least work in IT”. Him being a Software Engineer requires lots of studying and I know IT majors such as Vivek (The CEO I talked about) know numerous of different ethical codes, procedures, and programming languages. You don’t start of the first day as an engineer, you need to work your way up from an Intern or lower paid position. That is unless, you have a secondary degree. I want to be a Computer Engineer because it gives me the ability to express myself. I love programming and ever since being a little kid, I have always took apart of built computers. I also been making small apps and started off using Game Maker Studio or App Inventor and built my way up in a sense. I think Computer Engineering will give me a way to express a talent and hobby I grown to love.

Now in the next couple paragraphs I want to talk about Hardware and Software technologies. I want to talk about the major difference between Hardware and Software as well as the different implications. I want to also talk about how they are related in the field and are of both importance to me as a Computer Engineering student and Certified Technology Specialist. I want to talk about what it means to be a Hardware Engineer, a Software Engineer, and/or a

developer. I also want to give you an introduction to engineering in general and as a whole, while still keeping both an informative as well as formal tone. I hope you will enjoy.

One of the papers I decided to review that was related to my Uncle's profession was by Rex McElrath, it was entitled, "Software design example". The paper was fairly informative and the author used a Logos appeal to talk about many different facts throughout the paper. He informed us on basic definitions on Software design. He also has an ethos or character of being very technical with things and knowing proper terminology so it increases his credibility. Overall the author was trying to convey a message and that message was to teach us about standard definitions found in Software Design.

Some of the examples he gives are XML languages or Object Oriented languages in which he gives us proper abbreviations or syntax codes. This is a logos approach because programming is based on mathematics and critical reasoning. This is also an ethos approach because he knows the standard definition and has the language as well as characteristics of a credible author. One example of such syntax codes was found in page 26 where he used a standard xml based Java string to search for a document path. Throughout the paper, you will also see him talking about cases, legality, and different terminology that a Software Design Engineer should know in order to convey even more credibility.

This is in accordance with what my uncle Ameer have said about certain types of communications and protocols that are required by Software Engineering. As you can see it ties in with a classified standard ever Software Engineer knows. It also is related to Software modeling as well as following standard UML language procedures which is the standard language Rex used when making diagrams that were related to the message he was conveying.

Now in my next paragraphs I will review two more documents and tie it in with my Uncle's interview responses.

The next paper I will decide to review is called, "CE-Software Project Management", and its managed by T.M.G. Kleijkers. The paper also has more of a logos approach. It has an abstract in the beginning which follows academic standards in engineering as well as an index. It also talks about different stages of software quality assessment, and you can see that on page 9, it has a process model flow chart following UML communication standards.

This goes into what my uncle Ameer was saying or what the executives I talked about were saying. Engineering is a very difficult process and needs to be followed in a simple ethical standard. When you are a senior level engineer or a manager, you especially need to know certain procedures and roles in assessing the quality of your work. This author did a nice job explaining the terminology required in the CS Engineering process.

He also furthermore increases his credibility through an ethos-like process. This is done throughout his style of writing. He has a staff management plan as well as everything put up in an organizational way. He also knows proper work packages as well as ethical codes or levels of management in terms of the way he presented his paper. An example of his complex style of writing can be found in pages 12 to 14, in which he thinks critically in order to analyze risk.

The next document I will be analyzing is the Software Specifications IEEE template through a logos and ethos analysis. The logos analysis is that the paper is organized correctly in a way that follows correct engineering standards. The ethos analysis is that he knows the standard definitions of interfaces in terms of User, Software, and Hardware. This further emphasis his

credibility as he is following correct procedures in following the terminology and does it from an engineering perspective.

The overall character of the writer is viewed as someone who is fairly formal and organized in his work (considering it is just a template). He has overall done a nice job in terms of writing the paper as a thorough example for engineers. He also shows the revision history which furthermore emphasizes his credibility. Overall him, and many others go back to what my uncle said in terms of following different engineering procedures and writing through a formal aspect.

In engineering especially, you have to always write formally, structured, and organized. You will be doing many manuals, technical analysis sheets, flowcharts, and academic papers as an engineer. You are expected to learn how to write formally as well as in a manner to get both the technical side of the project as well as specify the project as a whole. It is important that you have credibility in what you write as an engineer, and organize your work. This is especially most important in analyzing your work and what you are turning in for your boss. That is why ethos and logos is fairly important as an engineer. Throughout this paper, I learned a lot on the formal writing of Computer Engineers and I hope you enjoyed.

Sources:

1. Interviews conducted with my uncle, Michael Nickola, and Vivek from Slidenerd
2. Sample documents by my professor which were different types of templates for design, specifications, and management report in software engineering

3. Rothwell, Edward J., and Michael J. Cloud. Introduction. *Engineering Writing by Design Creating Formal Documents of Lasting Value*. Boca Raton, Fla: CRC, 2014. N. pag.
Print.