This first week, I began my studies by doing the suggested reading assignments. Normally, I keep a note of the discussion and written assignment questions as I read, so as not to lose sight of important details to incorporate in to my writing from the readings. However, the readings did not seem to cover the discussion question directly, and as a consequence I did some independent research on the topic. I had never heard the phrase “legacy software systems” before. It was pleasant to learn through research than through reading and regurgitating a dry textbook.

Moreover, it was actually quite amusing and interesting to learn more about the topic. To be honest, it reminded me of having worked in a lab that had a VAX system sitting right next to a SUN workstation and wondering why the VAX system was still in service when a thumb drive had the same storage capacity. We also had tape drive storage (the big, refrigerator-size ones) with the same storage capacity of little LaCie drives the size of a book. And even in my home environment, I was holding onto an e-mac that was about 10 years old and refusing to upgrade any of the software on it because I knew it would cause issues. So, although I hadn’t heard the term “legacy software system”, I am fully aware of the phenomenon of lingering old technology!!

Another aspect of the reading material that I enjoyed this week is the learning modules. I very much appreciated their focused discussion with comprehension questions at the end. Being able to test my learning to see if I need to re-read something is very helpful and keeps me focused on the relevant material.

With regard to the lecture slides for the week, I was surprised and excited to see the case studies. I would not have thought of the gravity of the situation were it not for these examples.   
So often, when I think of software programming, I think of gaming or simple data crunching. Learning about the ethical realities of developing software for an insulin pump, patient records, and weather station was truly eye-opening and gave the discussion of ethics deeper meaning.

Having done all of the readings, I attempted the self-quiz. My score was pretty good (90%), but it is obvious I need to go back over the readings to review the generic software engineering framework. Having a way to test my understanding helps my meta learning quite a lot, so I know what I still need to work on.