What is software development? A question that I broached to myself after being asked if I like what I am doing and what path I have chosen. It’s a hard question to answer. The process and the the work is very foreign to people. The concepts and the things that I think about while working seem so small and unimportant to others who haven’t thought about them or more importantly haven’t had to sit and deal with the repercussions of not following best practices and as a result given birth to a crime of productivity that you then have to decide to either nurture into a product that you can work on or toss out and try again because the amount of time you will spend trying to turn that blob of work that you have created into something that can be molded and turned into something more than what it is now. These are impossible things to describe to others. These are also things that others do not care about. It has become a joke to talk about how little people want to hear about what you do since it isn’t tangible. The ideas, concepts, and builds aren't something that people can hold or see. The only person that holds them, sees them, and experiences them is you, the person who has created something either amazing that is very open to change or terrible and is impossible to add anything to no matter how much time you spend throwing code at the wall and seeing what sticks. That’s why every time I am asked while getting a haircut and the barber asks me what I do and I respond I am a software engineer they respond, “oh interesting”, and move on to ask me about literally anything besides that. I don’t think this is because people genuinely don’t care, it more that the process is scary. It is scary to myself, and hard to keep the drive to keep reiterating a bad creation into a beautiful one. Describing this process and these feelings to other people feels like trying to describe a generic to someone. Sure, in theory the essence is there. But the parts that really matter, or possibly the parts that people don’t get and I wish I could explain better or possibly more specifically I wish people would lend me an ear to let me flesh out… Make these parts none generic, fill them with substance. To do exactly that let’s break apart this generic class called software development and detail the processes that inherit it.

I would like to start with possibly the most important structure or class that the generic class of SoftwareDevelopment stands upon, Knowledge. This class provides the structure and base processes that all other sections of the SoftwareDevelopment class pull from.