Francesca Kang

CMSI 401

Johnson

Due: 30 Oct 2019

Pilone & Miles Book Assignment

1) The two major concerns of any software project are cost and time. From the customer's point of view, the ideal situation would be to spend the least amount of money to get their product as fast as possible. I believe that cost is a little bit more important than time, because it can be controlled. If the customer has almost unlimited funds, they could demand a large team to get their project done faster than a smaller team could. Time can't be altered to last longer for smaller funds. Complete functionality is the goal with a certain cost and time. The customer wants all of the features to function the way they want, however the amount of work that it takes to achieve said functionality could cost more or take more time than they originally thought.
2) The four main phases of the Agile method are shifts in focus that keep the main important idea in mind: 1) individuals and interactions 2) software 3) customer 4) response & change. I believe that if done properly, the 4 phases should be the most efficient way to iterate through each step of software development. Step 1 ensures accuracy for the step 2 that follows. Then step 3 is a double-check which matters most and step 4 "fixes" whatever is "wrong." When a lot of effort is put into each step, the phases don't have to cycle as often. For example, if step 1 and step 2 are done well, then the software is well designed and accurate to what the customer wants. So with a positive response in step 3, there is not a lot of changes left to do in step 4. Cycling through ensures that the final product is as perfect as it can get.
3) The main phases of the Waterfall method are: requirements analysis, design (product & system), code, test, maintenance. They flow once unlike the repeated cycle in Agile. I think they are very similar if requirements analysis and design are assumed to be like step 1, code like step 2, test like step 3, and maintenance like step 4, but I cannot be sure. It seems as though test might be a good requirement in step 2 of Agile that comes from

information in step 1 because it will reduce the amount of “wrong” things that the customer spots in step 3. Also, maintenance is different from fixing things that were pointed out by a customer. Maintenance is more of an automatic making sure that things are still working as they should be and updating to make it work. If an existing payment feature on a website has a payment feature from a 3rd party that has been compromised, the customer shouldn’t have to notice and point it out. The maintenance in Waterfall would automatically see that payment form isn’t working and change it.

4) A user story is description of a requirement from the customer’s point-of-view (for their users).

   Blueskying is an inclusive way of throwing out ideas for requirements.

   User stories should be: a 1) concise 2) customer-based 3) description of a requirement that serves the customer that 4) the customer can understand.

   User stories should not be: 1) a long essay that is 2) hard to understand (from customer’s POV) 3) and mentions what is going to be used to fulfill that requirement (technology).

5) Assumptions are not bad if they’re “safe.” Safe assumptions save time, but they are rare. A lot of safe assumptions are not noticed because they might seem like common sense. For example, a travel agency offers bus or plane options. It is safe to assume that a bus option should not appear for a trip from San Diego to Hawaii (impossible). For any other case that doesn’t deal with the obvious, assumptions are not safe. Generally, assumptions are not a good idea.

   A big estimate is inaccurate and should be reconsidered. It is hard to be certain about something so far ahead in the future. If a user story appears like its estimate needs to be big, then it probably needs to be split into smaller ones with the customer’s feedback.

6)

You can… User story	The more… User story	I help… Observation
I help… Roleplaying	In court... Observation	Some people… Estimate
Everyone’s involved… Blueskying & Planning Poker		

   I believe that the customer is technically involved in planning poker. Even though it is best to figure it out without the customer’s additional input, ideally the customer’s real wants and not assumptions of their wants are the basis of the estimates in planning poker.

7) A better than best-case estimate involves comparing different versions of estimates. Instead of focusing on trying to make the estimate as short or long as possible, the estimate's goal is to be realistic and accurate.

8) I believe that the best time to tell the customer that a delivery schedule won't be met is after a nice meal. Ideally, they would be in a good mood. Also, it should be as soon as possible, maybe before any work is started. This means the estimate given should have enough cushion so that once work is started, it fits the schedule. This conversation should have taken place after the "real" estimates (that include vacation, holiday, weekend, velocity) were done. It shouldn't be a difficult conversation because this is a time of going back and forth about the schedule and not a time when the customer is expecting that work to be done soon.

9) Organized branching is a good idea because it makes a record of software versions while maintaining their relationships to each other. This only works for "whole versions" of the same part. For example, a proper branch should be optimized visa acceptance from plain visa acceptance not mastercard acceptance from visa acceptance. This way, the "original" is kept and the "real" (better/updated) version is clear.

10) I am not completely familiar with any specific build tool, but I do know that they can make developing faster. They are a little bit like shape stencils for programming. They can make code neater and provide basic parts of code. For example, having a triangle and rectangle stencil makes drawing a house faster and neater. However, the person still needs to fill in all the details that make it an actual house. If the build tool isn't the right one for the project, then it's not going to be helpful the same way an octagon stencil isn't helpful for drawing a rectangle bed.