Thomas Ochsner

10/30

1) How much will a project cost and how long will it take are the two main concerns for any project. And I think time is more important than money because a lot of the time it is more limiting. Throwing more money and people at a project will not speed it up, but if you have a project that simply needs more money and is on track then investors could come in or the team could take a temporary pay cut. Knowing how much time and money a project will cost is instrumental in a smooth development cycle.
2) With agile software development you get planning, analysis, design, coding, and testing. If you want a good complete job done then everything needs to be done again even when just updating an existing app. The difference between the iphone 1 and the iphone X is pretty significant. For a project to be successful, I think it needs to go through all stages.
3) In the waterfall method the main phases that occur our Requirement Analysis, Product Design, System Design, Coding, Testing, and Maintenance. This has more steps than agile development and it includes maintenance at the end so something that is a live service and requires constant oversight and fixing might work better under this dev plan. A team of developers that are farmed out to

make an app and then not touch it again would be less inclined to follow this method.

4) a) Describes a type of user, what they want, and why. b) Similar to big sky thinking and thinking outside the box, it means that your ideas are not being constrained. c) User stories help explain the target demographic for a project d) User stories should not be too specific because then it will be too hard to derive meaning from that example.

5) a) Not true at all. Sometimes assumptions are very basic and can only lead to a small amount of confusion with a worse case scenario. Also, sometimes an individual is very knowledgeable in a field and their assumption is accurate. b) Once again I don't agree. A big user story covers a lot of ground and might be inaccurate but it can help someone else then understand smaller user stories that can be derived from that one big user story.

6) a) User Story b) User story c) Blueskying d) Role playing e) Observation f) Observation g) Estimate h) Blueskying

   Got all the same as the book

7) It is an unrealistic estimate in which everything goes better than expected for the project.

8) It is best to tell them as soon as possible so that they have better expectations. It gives you more time to adjust how you need to and allows for more honest negotiations back and forth for time and money.

9) It is good. If you have a version that breaks beyond repair and you need something to show a client then going backwards to another working branch will help your cause.

10) I have used a build tool in development. The good thing about it was that I could quickly see my progress and what the updated project looked like. The bad thing is that you will not get as much feedback.