Dreaming in Code Questions

Overarching Themes

Pay attention / record the various roles that software engineers have on the Chandler Project.

Pay attention / record the many scheduling issues related to Chandler Project.

Chapter 0

1. Who wrote "software is hard?" Who is that guy?
    - Donald Knuth, author of The Art of Computer Programming.
2. Programmers start counting at what number?
    - 0
3. What was the original sense of a "hacker?"
    - Obsessive programming tinkerer.
4. According to a 2002 NIST study what % of software came in significantly late, over budget, or was canceled?
    - 2/3
5. Who wrote the 1987 essay entitled "No Silver Bullet?"
    - Frederick P. Brooks Jr

Chapter 1

1. What roles in the Chandler project did Michael Toy, John Anderson, Ted Burgess, Mitchell Kapor, and Lou Montulli hold.
    - Toy: Project manager.
    - Anderson: systems architect.
    - Burgess: programmer.
    - Kapor: Founder and funder.
    - Montulli: programmer
2. What is "Bugzilla?"
    - Bug tracking software.
3. What is OSAF?
    - Open Source Applications Foundation
4. What is the projects name?
    - Chandler
5. What will the software do?
    - Personal Information Manager
6. What is Toy's keyword for "black hole" bugs?
    - Scary
7. What scared Toy so much about Bug 44?
    - Not knowing the time it would take to fix it.
8. What did Toy refer to as a "snake?"
    - Important problem with no consensus on how to attack.
9. In the software world, what does "slippage" mean?
    - Missing project milestones.
10. Fredrick Brooks was a programming manager for what software project?
    - IBM System/360.

11. What is Brooks' Law?
    - Adding manpower to a late software project makes it later.
12. Brooks found what % of project time was spent writing code?
    - 25%
13. Brooks found what % of project time was for testing and fixing bugs?
    - 50%
14. Brooks observed that the unit of effort named “man-month” only applied under what conditions?
    - Where tasks were divisible to the point of not needing to communicate.
15. What is the difference between source code and the program you install (.exe) on your computer?
    - Source code has the instructions for the compiler. Binary programs are indecipherable 1’s and 0’s for (most) humans.
16. What is the one “article of faith” that all “open source” or “free” software advocates share?
    - Open source sofrware will improve in ways that closed source cannot.
17. What is the difference between a “good” programmer and a “great” programmer?
    - Good programmers know what to write. Great programmers know what to re-write.
18. Eric Raymond’s book “The Cathedral and the Bazaar” made a distinction between two important project development ideas, briefly contrast them.
    - Cathedral being cloistered and ‘closed’ source. Bazar being an open market, laid bare for all. They compare the model of commercial software (e.g. MS Office) and open source (e.g. Libre Office).
19. Has “open source” software project development refuted Brooks’ “mythical man-month” concerns?
    - No, it is simply ‘an alternate universe of programming’ with different timelines.
20. What was Andy Hertzfeld’s input when the Chandler project appeared to have stalled?
    - Stop designing and start coding.

Chapter 2

1. What was the lifetime as a supported product, of Lotus 123? When did Kapor walk away from it? Why did he walk away from it?
    - Lotus 123 was launched in 1983 and ceased support in 2013. Kapor quit the company citing ‘intolerable personal conflict’.
2. What does it mean for a program to “fork?”
    - To fork is to split into two separate programs with different goals.
3. Linus Torvalds used a “science” and “witchcraft” analogy referring to software, explain.
    - Science being open, transparent, and built on top of people’s previous work. Witchcraft is like a secret, not being shared.
4. Who, where, when demonstrated one of the first PIM software programs?
    - Engelbart, 1968 San Francisco Convention Center.
5. People often refer to starting their computer as “booting” their computer. What was the origin of this term?
    - On older computers, one needed a bootstrap loader program to start the OS.
6. Where was the graphical user interface (GUI) developed?

  - Xerox's Palo Alto Research Center.
7. List three software project "train wrecks."
    - The FBI's Virtual Case File program.
    - US Army's Future Combat System.
8. Scan down this article to the two conclusions (about eight paragraphs down). With two sentences, what is your take away from this? http://scribblethink.org/Work/Softestim/softestim.html

    - Software timeline estimation is difficult, impossible from an objective standpoint. Ethically, projects should not be promised on objective timelines, because programming is subjective and difficult to quantify productivity.

Chapter 3

1. Keep track of team members: 2001 --> Mitch Kapor, Morgen Sagen, Al Cho, Andy Hertzfeld
2. When introducing a new technology or design, why did Frederick Brooks advise "plan to throw one away?"
    - Because you almost certainly won't get it right the first time.
3. 2002, Katie Parlante joined Chandler. When she first joined what did she do?
    - Researched user behavior patterns.
4. John Anderson joined Chandler as "systems architect." Briefly, what did that mean?
    - Systems architect means that he has final say on the matters relating to design and coding.
5. What is a "core" dump? Why the use of the word core?
    - When a computer drops everything and spits out an output of everything in it's memory. The core is a physical part of the computer that controls functions.
6. How did Erik Sink, speaking of abstractions, describe what programmers do?
    - They build piles of abstractions.
7. Rather than writing program statements in binary code, 110101110 1001101111, programmers developed a shorthand language called what?
    - Assembly language.
8. Adding layers of abstraction, new programming languages were created: Lisp, Cobol, Algol, Basic. Fortran was the first widely used. What kind of program converted Fortran to binary?
    - Compiler.
9. What are the implications behind: "...there is no Moore's Law for software. Chips may double in capacity every year or two; our brains don't."?
    - Humans will not keep up with computing power.
10. The language selected for Chandler was Python. What does it mean that Python was an interpreted language?
    - Python is ran from source code to a translator then to a processor.
11. The announcement of Chandler occurred when? (Interesting Paper) Slashdot page from 2002.
    - 10/20/2002

Chapter 4

1. More programmers join the Chandler team...
   Lou Montulli, Aleks Totic (volunteers): experience with internet protocols (Netscape)
   Chi-Chao Lam: Product Manager
   Pieter Hartsook: OSAF's marketing and PR
   Jed Burgess: GUI, David McCusker: data storage
   Michael Toy: Programming team manager
   Andi Vajda: OSAF's repository developer
   Brian Skinner (volunteer)
2. What do "front ends" and "back ends" mean to software developers?
3. The terms RAP, SQL, object persistence, all relate to a problem Chandler's developers needed to solve at the earliest stage. What was the problem?
4. What did the Lego Hypothesis refer to?
5. Give one reason why the Lego Hypothesis seems to not work so well.
6. How many "modules" does CPAN currently host?
7. After Chandler was announced in Oct. 2002, when was the first public release?

Chapter 5

1. What is the three-way trade-off that many software projects struggle to overcome.
2. Michael Toy: Software development manager, does what?
3. What is the more recent definition of "geek?"
4. Kaitlin "Ducky" Sherwood: Help with Chandler's approach to email, organize project documentation. Important?
5. What does "refactoring" mean to programmers?
6. "In software management, coordination is not an afterthought or an ancillary matter; is the...." Finish the sentence.
7. What is "yak-shaving?"

Chapter 6

1. What is term "edge cases" referring to in software development?
2. What did M. Kapor argue that software design was? (One sentence.)
3. Montulli and Totic leave the project
4. Heikki Toivonen, Stuart Parmenter, and Mimi Yin are hired to do what?
5. Andy Hertzfeld leaves
6. Michael Toy leaves
7. Mitch Kapor takes over overall management of development
8. Summarize briefly Linus Torvalds advice about "large projects" give in 2004

Chapter 7

1. Ted Leung hired to do what?
2. Donn Denman hired (2004) to do what?
3. Briefly describe Hungarian notation
4. What does the author state is the "...single most challenging demand of software development."
5. Brian Kirsch, David Surovell, and Lisa Dusseault joined the effort in what roles?
6. Katie Parlante assumes mantle as "architecture coordinator", what is this?
7. Lisa Dusseault takes on "detailed planning and scheduling of the development work." Any idea what this is?

Chapter 8

1. What does "eat your own dogfood" mean?
2. What happened to the "peer-to-peer" sharing ideal in Chandler? (This is related to the WebDAV discussion.)
3. Quote: "When people ask for numbers that far out, the traditional thing that engineers do ...." When discussing the timeline for Chandler, how was the quote above completed?
4. Aparna Kadakia became quality assurance (QA) manager. Sheila Mooney joins Chao Lam on product management writing more formal and detailed specifications.
5. As "reality" set in, what happened to Chandlers goals?
6. Brian Skinner leaves, Stuart Parmenter leaves, Chao Lam leaves, and Ducky Sherwood leaves.

Chapter 9

1. After about two years with little to show for it, the author suggests the project had something else important. What?
2. Structured programming evolved to address what programming practice?
3. Was structured programming a solution to the problem of software development?
4. Have any techniques shown to improve the software development process?
5. Give a few software failures the author lists. Did anyone die due to the errors?
6. The "waterfall model" of programming was/is popular. What were some problems with this model?
7. What are the four tenets of Agile Software Development?
8. What did a 2004 study find about the development practices of some two hundred software team leaders?
9. What is the "Joel Test" and what did he say about Microsoft and the Joel Test.
10. What is Rosenberg's Law?

Chapter 10

1. Chapter 10 is about the notion of "Software Engineering" and the difficulty of applying this label to the development of software. The author suggests that Yertle the Turtle provides an important lesson for programmers. Describe it.

Remaining Pages

Complete the reading reflecting on the Chandler Projects **scheduling** issues and the various **project roles** that were important on the project.