

Chapter 18: A practical Mechanism for Ethical Risk – Assessment – A SoDIS Inspection

Amazon Link : http://www.amazon.com/Handbook-Information-ComputerEthics/dp/0471799599/ref=sr_1_1?ie=UTF8&s=books&qid=1269175801&sr=8-1

Quote: “Sometimes we find the cost/benefit analysis of this form of risk analysis troubling”

What I expect to learn: What I expect to learn from this chapter is the definition and what is a practical mechanism all about and its relation to ethical risk and the assessment for A SODIS inspection.

Review: The chapter talks about A practical mechanism for ethical risk but let us first define what practical mechanism is all about. Practical Mechanism is the study for the simplest way of the modern world today it is applied in the information technology through machines and other things that is related to information technology and should consider if it's ethical or not. Today many software's are being developed but not all this software's are surely developed because there are possible things that might happen this is the number one problem of developers because sometimes they are finished developing a software and while testing the software they will encounter some minor problems that has to rebuild and redevelop all again because the whole code or thing is in a one source code but with the technology getting more modern there are actually programming languages were they have shortcuts and it won't be hard for developers to develop a software because names of codes are already in the programs but some programmers or developers prefer the old version without these automatic names because they might not memorize all this things and they might not learn something from it because they will be dependent on the word that is appearing and will keep them from memorizing terms which is a bad thing and developers should be careful and consider if what or if the software they are developing is good or bad and unethical or not because this software might be rejected and not be recognized by public users and professionals but are hackers who are used to these kind of things.

What I have learned:

1. Developed software's still has problems.
2. Example of these is the apple or Mac which still has some problems and the windows 7.
3. Developers should double check the software they are developing to avoid future problems and complaints from the users.
4. When developing software developers should consider the risk.
5. They should analyze and plan the whole software development properly.

Integrative Questions:

1. What is the relation of the chapter to ethics?
2. Is the continuous development of software's like upgrading software from 0.5 to 0.6 and another one a good thing?
3. What are the possible solutions to the problems that might occur while developing software?
4. What should the developers consider in the way users will treat this or is it a user friendly?
5. What is the situation where can software be unethical, in which way?