Book Questions
Chapters 10
Devayan Mandal

10.1 Suggest six reasons why software dependability is important in most sociotechnical systems.

Ans: The following are some of the many reasons why software dependability is important in most sociotechnical systems:

1. A software system which is not fully dependable can put the lives of people at risk. For example, if an airline monitoring software system does not provide accurate information on a timely basis, it can lead to many consequences or even accidents.

2. A software system, which has to do with management of finances can greatly affect the correct procedures used during buying, selling and other modes of transaction if there is any loss of precision or dependability in the system.

3. Almost every software system is required to meet certain quality control specifications before it is released commercially. The quality control requirements may be specific to the type or classification of the software system. Failure to do so makes the system undependable for use and may receive a poor rating from companies like the Bureau of Better Business (BBB).

4. A software system which is not dependable and does not meet the system requirements may damage the hardware systems which it is running on.

5. This in turn, can affect the security and stability of valuable information or data which is stored in the hardware. For example, overheating of the hardware or excessive demands placed on the system may lead to a hard-drive crash.

6. As we observe, the cascading effect may continue and the company or users who use this software system will eventually avoid using the system. If the company who developed the system has other products on the market, end-users may hesitate to use those too because of the poor user experience.

10.2 What are the most important dimensions of system dependability?

Ans: Important dimensions regarding dependability for a software system:

a. The software system should provide an appreciable degree of security from external threats.

b. The software system should be designed in a way that it does not damage the end-user's hardware or lose sensitive data in the case of a system crash. A 'safe mode' of operation should be provided to ensure that important components of the system can be run.

c. The software system should not take excessive time to start up, run and operate compared to other similar competing systems on the market.

d. The software system should be able to win the end-user's trust and ensure that it accomplishes its tasks in accordance to the user's requirements.

10.6 Explain why it is reasonable to assume that the use of dependable processes will lead to the creation of dependable software.

Ans: The use of dependable processes usually leads to the creation of dependable software. If a certain process or component of the software system is deemed dependable during several process runs or operations, it should be expected to provide the same reliability during future operation(s) of the system. Also, if the system has proved itself to be dependable, any employee who is trained to operate the system can do so and expect correct system operation and execution.

10.8 Give two reasons why different versions of a system based on software diversity may fail in a similar way.

Ans: Different versions of a system based on software diversity mail fail because:

1. If the different versions of the system have the same runtime error. For example, if the team which initially developed the software system left an error unattended and if the next team, which developed a version update to the software left the error unattended too (possibly because they were not made aware of it), then the failure or runtime error would be present in each of the software system versions.

2. The quality analysis stage of development did not adequately match aspects of the system developed by the software team to the specification requirements. Such a scenario may not translate to the desired output for the end-user even though no errors are produced by the written code.

If such an error is left unattended, there is a high probability that it be left unattended during an upcoming version update.