

# Check point systems development life cycle (sdic) – shortcut

[Life](#)



**ASSIGN  
BUSTER**

A problem can be defined as the difference between things as perceived and things as desired. A difference does not exist between what you perceive you have and what you desire to have, then there is no problem. In our process, we first need to understand if there really is a problem. This definition of a Problem; is different from many other ways to define the word Problem. TTT addresses the gap between our perceptions of what our customer wants and what they actually desire. One advantage of this definition is that it emphasizes that there are many ways to solve a problem: CLC Change the customer's perception of what they have now.

For example, many of the requests for enhancements that Microsoft gets are for features that already exist. CLC Change the customer's current desire. For example, if the customer finds that what they want will cost millions of dollars over their budget, then maybe they will scale back their desires and focus on solving a simpler problem. C] Change the gap between what customers want and what they actually desire. A software solution is usually an attempt to narrow the gap. Problematically Steps: 1. Identify the old problem. 2. Understand the root causes. 3. Gain agreement on the problem definition. 4. Identify constraints on the system or project. 5. Identify and validate the solution against the root causes. 6. Define the solution systems or UN d ar y. Above is a list of activities that you go through when performing problem analysis. Start by identifying stakeholders for the problem. These stakeholders are people who are affected by the problem. Next, understand the root causes for the problem (the real problem behind the problem). Once the real problem (root cause) has been identified, get all your stakeholders to agree to a description of the problem.

Armed with a description of the problem, identify any constraints that may be imposed on any solution. These constraints typically limit the solution you may provide. Next, identify possible solutions for the problem. Let it be important that you identify more than one possible solution. By doing this, you can then compare and contrast the solutions to find the best possible solution. Phase one is identifying problems, opportunities, and objectives which requires that the analyst honestly analyze what is occurring in a business.

The analyst should check aspects of information systems by addressing specific problems or opportunities. The analyst should be able to see if some aspects of information systems applications can help the business reach its objectives through addressing specific problems or opportunities. Phase two is determining information requirements. The analyst strives to understand what information users need to perform their jobs. Phase three is analyzing systems needs. Analyze system needs by creating data or process flow diagrams, document procedures, and etc. Phase four is designing the recommended system.

The systems analyst uses the information collected earlier in order to accomplish the logical design of the information system. Phase five is developing and documenting software. This works with users and communicates to the developers what needs to be programmed and documented. Phase six is testing and maintaining the system. A series of tests to pinpoint problems is run first with sample data and eventually with actual data from the current system. Phase seven is implementing and

evaluating the System. This is reviewing and evaluating the system. This also involves training users to use the system.