

Course Learning Outcomes for Unit VIII

Upon completion of this unit, students should be able to:

1. Analyze the key differences between data, information, information technology (IT), and information systems (IS).
2. Explain the similarities and differences of personal knowledge and management tools.
3. Examine the importance of mobile systems with regard to securing information and knowledge.
4. Explain how information systems can be used to gain and sustain competitive knowledge.
5. Evaluate the approaches to developing organizational knowledge management strategies.
8. Evaluate major types of hardware and software used by organizations.

Course/Unit Learning Outcomes	Learning Activity
1	Unit VIII Essay
2	Unit Lesson Chapter 12 Unit VIII Essay
3	Unit VIII Essay
4	Unit VIII Essay
5	Unit Lesson Chapter 12 Unit VIII Essay
8	Unit VIII Essay

Reading Assignment

Chapter 12: Information Systems Development

Unit Lesson

Business Processes – The Software Development Life Cycle (SDLC)

Information technology (IT) projects can fail for a number of reasons such as missed deadlines, failing to meet requirements, lack of upper-management support, and exceeding the projected budget. Using a development method can help reduce the risk of project failure. One of the development methods that system developers use is called the software development life cycle, or SDLC for short. The SDLC method is also known as the waterfall method because it has five phases; each phase is completed in sequence and serves as a framework for developing an information system (IS) or an IT project (Figure 1) (Kroenke & Boyle, 2017).

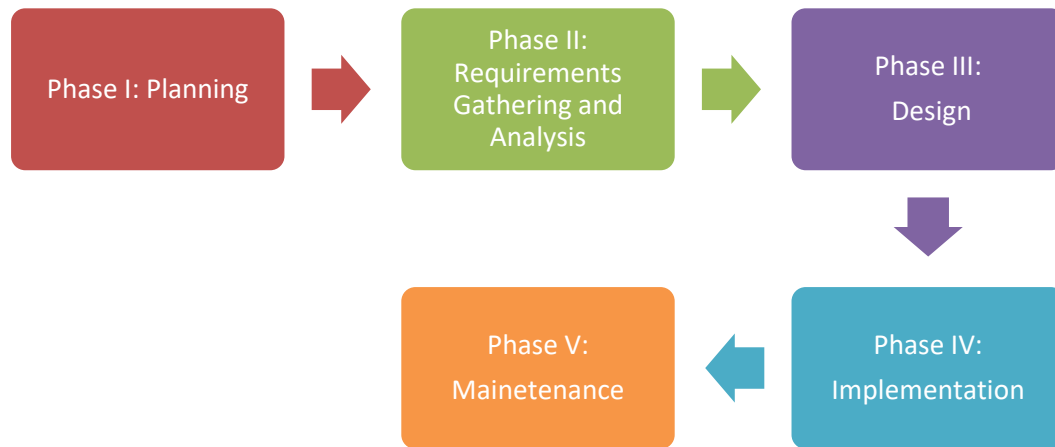


Figure 1: The five phases of the SDLC process
(Kroenke & Boyle, 2017)

The first phase—the planning phase—is a very critical part of the SDLC process because, at this point, the system’s designer or developer must define the problem that the organization is facing. Common concerns are the lack of competitiveness with other organizations in the industry. For example, management might voice concerns about the lack of efficiency for an inventory control system because the system is outdated (Kroenke & Boyle, 2017).

In the planning phase, it is helpful to remember the four “Ws”—who, why, when, and what.

- *Who*: Who will be the main users of the system?
- *Why*: What problem will the system solve?
- *When*: When will the system need to be implemented?
- *What*: What will the system do (Kroenke & Boyle, 2017)?

Testing System Scenario – Phase I

In a recent management meeting, several managers voiced concerns about the high rate of project failure when implementing new systems. The organization has a facility for testing systems before they are deployed, but there is no formal process. It is this process that managers believe is the reason for the high rate of system implementation failure. For example, a new database system was implemented, but several users complained that their system would crash when using a certain feature.

In phase I of the SDLC process, system developers will need to address the four “Ws”—who, why, when, and what.

1. *Who*: The main users of this system are applications testers, QA testers, systems engineers, security analysts, and software developers.
2. *Why*: The new system will provide a way to process and track the development and testing of new systems before they are implemented.
3. *When*: The system will need to be operational within the next 6 months before the next IT project is complete.
4. *What*: The new system will solve the testing process problem that appears to contribute to the failure of IT system implementations.

The second phase—requirements gathering and analysis—is where the designer or developer must identify and define the problem that needs to be solved. From this information, they can then develop the requirements for the system and analyze those requirements to determine if a new system will solve the problem with the current system. To analyze the problem and determine the requirements for the new system, system developers will need to use techniques for gathering the necessary information for the system. Some examples of information gathering are interviews, observations, surveys, or a joint application design (JAD) system. A JAD system is where users, management, and IT professionals gather in a workshop to discuss

and develop an application. The JAD approach incorporates various viewpoints so that the application is developed according to these requirements (Kroenke & Boyle, 2017).

Testing System Scenario – Phase II

A JAD session was held where various personnel involved in the systems testing process were present. The system requirements were identified, and these are listed below:

- how the users would use the system,
- how they currently use the system,
- what problems were present and how the system should address these problems,
- what the users expect from the system,
- how decisions will be made and the data that will be needed to make the decisions,
- where the data will come from,
- how the data should be presented, and
- what tools should be made available to the users to help make decisions.

A formal workflow design diagram was developed to help develop the requirements (Figure 2).

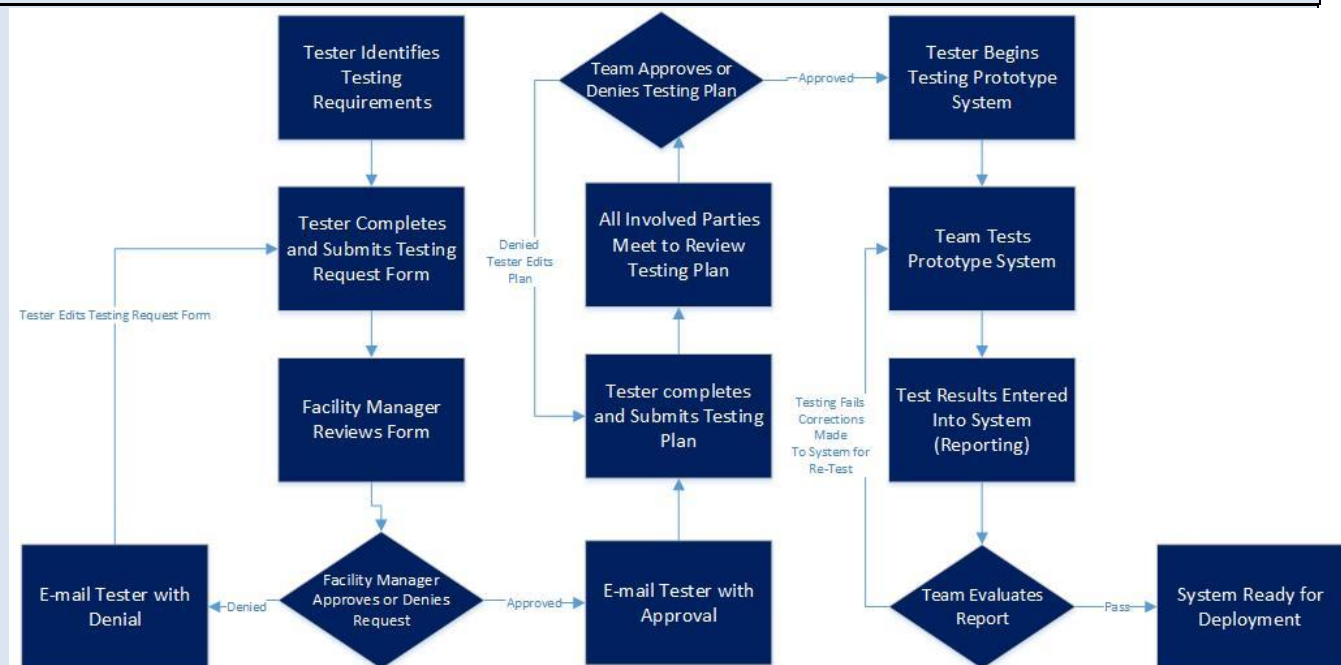


Figure 2: Pictured is the testing process flow for system implementation. The workflow diagram clearly shows how data will flow between processes, external entities, and data stores.

In the third phase—design, a solution is defined for an IS that will best solve the problem that the organization is facing. Generally, this solution is outlined and documented with the requirements and specifications of the new system. This documentation can be in the form of diagrams, databases, files, forms, and reports. These documents should provide details about the hardware, software, processes, networking components, and any other system specifications (Kroenke & Boyle, 2017).

The design phase is completed in three parts: conceptual design, logical design, and physical design. The conceptual design is generally an overview of the system that details how it will work. This can be done through the use of flowcharts, computer-aided software engineering (CASE) tools, or models. The second part is the logical design where more specific information is provided such as the hardware and software that will make up the system. The third part, physical design, is where a graphical representation of the system is presented. This includes the hardware, networking components, and other technical specifications. This is also where the final design/specification document is developed to assist in the final stage of the process (Kroenke & Boyle, 2017).

Testing System Scenario – Phase III

Now that the team has determined how the system should work, a conceptual design should be developed to illustrate the workings of the new system (Figure 3).

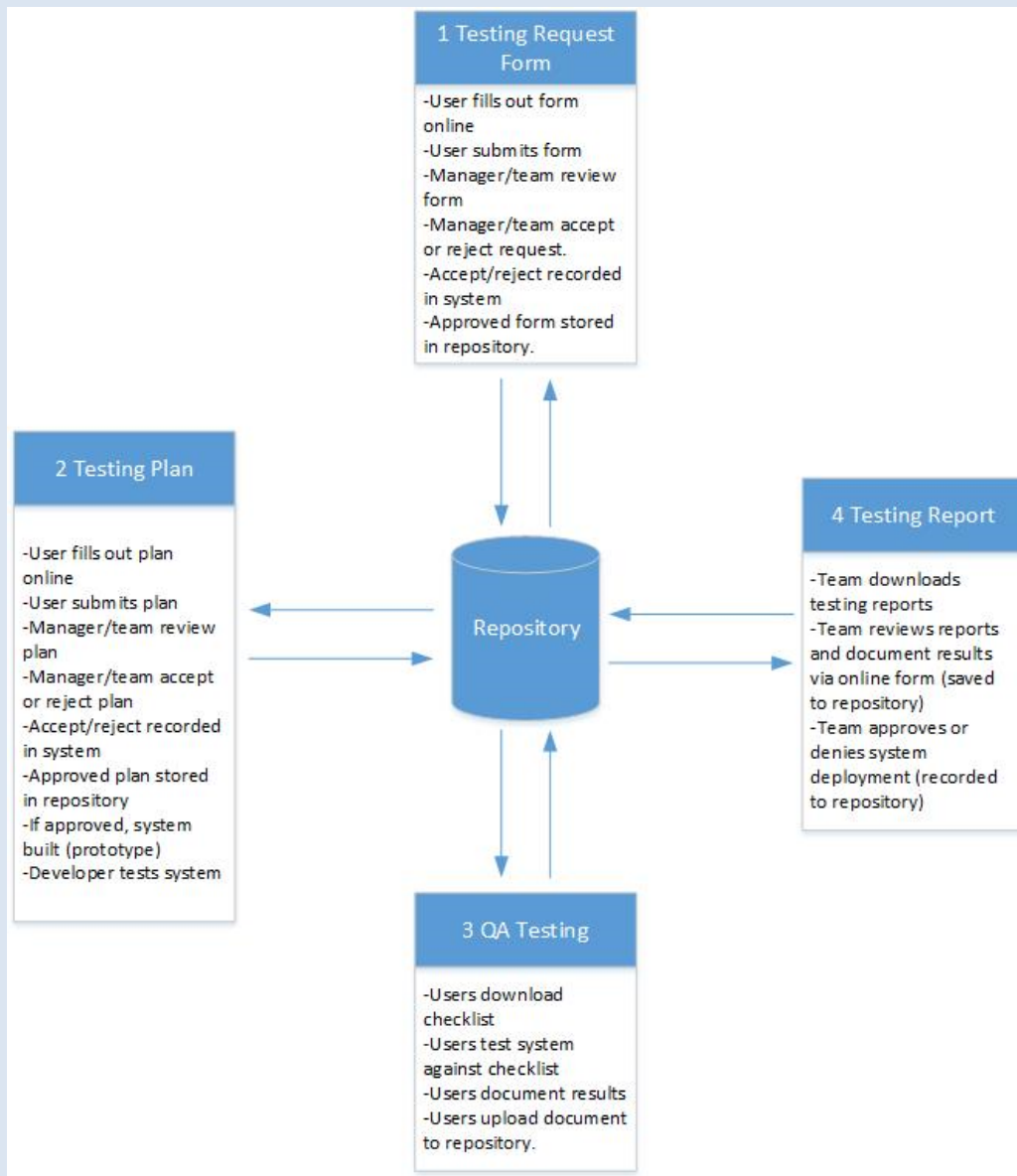


Figure 3: Pictured is a conceptual model example. Next, a logical design should be developed to illustrate how the system will work, including the hardware, software, and networking components.

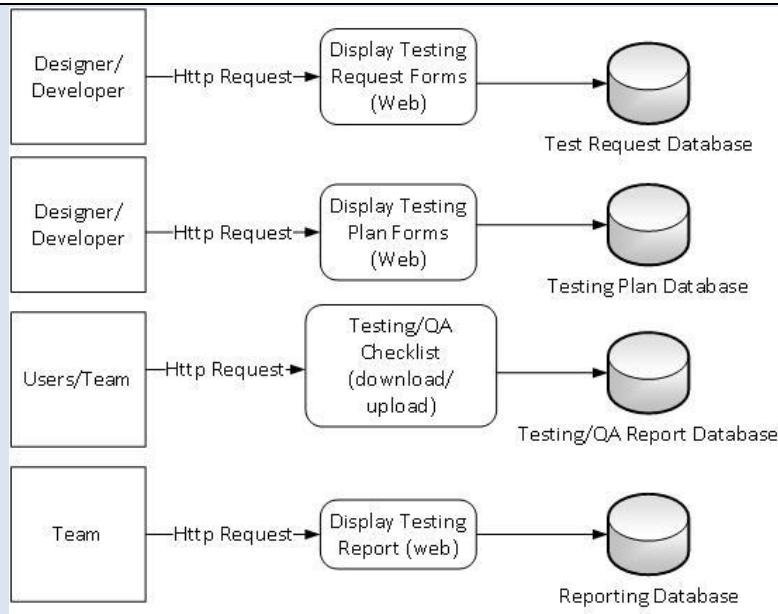


Figure 4: Pictured is a logical design example. Lastly, the physical design should be developed to explain the physical components and interaction.

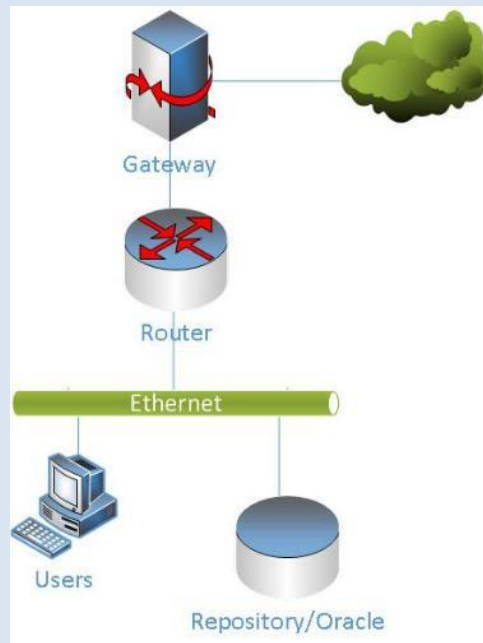


Figure 5: Pictured is a physical design example.

In the next stage—implementation, the designs from phase II are implemented. Components for the system, such as hardware, software, and labor, are procured, and the system is configured. In addition, employees are trained, the system is tested, security measures are implemented, documentation is created, and a disaster recovery plan is developed (Kroenke & Boyle, 2017).

Testing System Scenario – Phase IV

Now that the design phase has been completed and the final design approved, the team moves forward to put the system in place.

1. Procurement Team

- a. Procure all hardware and software per the design/specifications document.
- b. Inventory all procurements.

2. Design/Development Team
a. Put together the system including the software and networking components. b. Document processes, and record changes to the design/specifications document.
3. Testing Team
a. Test the system according to the QA checklist. b. Resolve any issues, and record events to the design/specifications document.
4. Training Team
a. Train users of the new system. b. Provide documentation as needed.
5. Security Team
a. Implement security scans and set user permissions. b. Resolve any issues and record events to the design/specifications document.
6. Deployment Team
a. Deploy system.

The final phase—maintenance—is probably the most undervalued process in the SDLC model. After the system has been implemented, there is still more work to do. In this phase, the maintenance team assesses the overall functionality of the system. This can be done by analyzing performance data and by talking with users to see if the system is meeting their expectations. If any issues or problems are encountered, the maintenance team should take action and resolve them. One way to accomplish this is to establish a help support system such as a help desk or to train a few technical support personnel on the system so they can provide support to users. If the team finds that the system is not performing as expected, it may be necessary to return to the planning phase and start all over. This is why it is important to not only follow a systems development method but to also have well-designed phases of development (Kroenke & Boyle, 2017).

Testing System Scenario, Phase V

The system has been deployed, developers and testers are using the system, and several IS projects have been completed using the new system. Now, it is time to think about supporting the users and evaluating how well or not so well the system is working.

1. Perform system metrics on a monthly basis.
2. Interview system users 1 month after deployment.
3. Identify any system issues, and report them to management.
4. Resolve system issues.
5. Setup a support system for users (help desk, documentation).

Summary

The SDLC methodology is only one of several development processes that can be used for developing IS or IT projects. Examples of other development models are service-oriented architecture (SOA), rapid application development (RAD), extreme programming (XP), pair programming, and agile methodology (Kroenke & Boyle, 2017).

The type of method used will depend on the organization and the type of system or project being developed. For example, one would not use the SDLC method if the user's needs keep changing, the system requirements keep changing, the scale of the project is too large or too small, personnel does not have the expertise, or the system specifications have not been fully defined.

The SDLC can have more than five phases; for example, one can add an additional phase for testing or software installation, depending on the type of project. While some phases of the SDLC may be more difficult than others, none of these phases should be overlooked. Any oversight could cause the system to fail to meet expectations or perform poorly.

Reference

Kroenke, D. M., & Boyle, R. J. (2017). *Using MIS* (10th ed.). New York, NY: Pearson.

Suggested Reading

In order to access the following resource, click the link below.

Inventor Jim Holley discusses the challenges in creating his first product and the difficult process of inventing his product. He explains how to get a prototype of a product. If needed, a transcript is provided once the video is accessed.

Business2Learn. (2011). *Entrepreneur's journey: Making a prototype* (Segment 4 of 21) [Video file]. Retrieved from <https://libraryresources.columbiasouthern.edu/login?auth=CAS&url=http://fod.infobase.com/PortalPlays.aspx?wID=273866&xtid=47860&luid=138513>

Learning Activities (Nongraded)

Nongraded Learning Activities are provided to aid students in their course of study. You do not have to submit them. If you have questions, contact your instructor for further guidance and information.

To test your knowledge of the material covered in this unit, complete the activities listed below.

- Chapter 12 Discussion Questions
- Chapter 12 Active Review
- Chapter 12 Using Your Knowledge
- Chapter 12 Collaboration Exercise
- Chapter 12 Cards

The activities are located within the chapter readings in uCertify. The Chapter 12 Discussion Questions, Active Review, Using Your Knowledge, and Collaboration Exercise are located at the end of the chapter. The cards can be accessed by clicking on the Cards icon within uCertify, which is located to the right of the chapter title, and the icon in uCertify resembles the image shown below.

