

CHAPTER 2
SYSTEM DEVELOPMENT LIFE
CYCLE

After completing this lesson, you will be able to:

Define a system

Explain System development Life Cycle and its importance

Describe objectives of SDLC

Describe stake holder of SDLC and their roles

Explain the following phases of SDLC

- Planning
- Feasibility
- Analysis
- Requirement engineering
 - Requirement gathering
 - Functional requirements
 - Non-functional requirements

Requirement validation

Requirement management

- Design (algorithm flowchart and pseudo code)
- Coding
- Testing/verification
- Deployment/implementation
- Maintenance/support

Explain the role of the following in SDLC

- Management in SDLC
- Project manager
- System analyst
- Programmer
- Software tester
- Customer

2.1 SYSTEM DEVELOPMENT LIFE CYCLE

Q1. What is meant by Systems Development Life Cycle?

Answer

The Systems Development Life Cycle (SDLC), in software engineering is the process of creating or altering information systems. In other words these are the models are methodologies that experts use to develop these systems.

Q2. What is software engineering?

Answer

Software Engineering

Software engineering is an engineering approach for software development. In software engineering, the SDLC concept reinforces many kinds of software developments techniques. These techniques form the framework for planning and controlling the creation of an information system.

2.1.1 A SYSTEM

Q3. Briefly explain the term 'system'.

Answer

System

A system is a set of components (hardware and software) for collecting, creating, storing, processing and distributing information. A system can be developed by applying a set of methods procedures and routines in a proper sequences to carry out some specific task.

2.1.2 SYSTEM DEVELOPMENT LIFE CYCLE (SDLC) AND ITS IMPORTANCE

Q4. State the purpose and importance of System Development Life Cycle.

Answer

System Development life cycle (SDLC)

System development life cycle (SDLC)) is a problem solving process through which a series of steps or phases helps to produce a new computer system

Purpose of System Development Life Cycle

The basic purpose of System development life cycle is to develop a system in a systematic way in the perfect manner.

Importance of System Development Life Cycle

1. System development life cycle (SDLC) delivers quality software, which meet the system requirements
2. System development life cycle (SDLC) ensures that the requirements for the development of the software system are well defined and subsequently satisfied.
3. System development life cycle (SDLC) delivers cost-effective system
4. System development life cycle (SDLC) maximizes the productivity.

2.1.3 OBJETICES OF SDLC

Q5. Describe the objectives of System Development Life Cycle

Answer

Objectives of System Development Life Cycle

A systems development life cycle (SDLC) has three primary objectives

1. Ensures that high quality systems are delivered
2. Provide strong managements controls over the projects
3. Maximize the productivity

Summary of Objectives

These objectives are summarized as follows

1. One of the major objectives of SDLC is to establish an appropriate level of management authority to direct, coordinate, control, review and approve the software development project.
2. SDLC should identify the potential project risks in advance so that proper planning should be done in early.

2.1.4 STAKEHOLDERS OF SDLC

Q6. Describe stakeholders of SDLC.

Answer

Stakeholders of SDLC

Stake holders of SDLC are those entities or groups, which are either within the organization or outside of the organization of that sponsor, plan, develop or use a project. Stakeholders may be users, managers and developers. It is the duty of the project management team to identify the stakeholders determine their requirements, expectations and manage their influence. In relation the requirements to ensure a successful project.

2.1.5 SDLC PHASES/STEPS

Q7. Explain the phases/steps in SDLC.

Answer

Phases/Steps in SDLC

The following are phases/steps in SDLC. These phases are shown in Figure 2.1



Figure 2.1: Phases of System Development Life Cycle

- i. Defining Problem
- ii. Planning
- iii. Feasibility Study
- iv. Analysis
- v. Requirement Engineering
- vi. Design
- vii. Coding
- viii. Testing/ Verification
- ix. Deployment/ Implementation
- x. Maintenance/ Support

1. Defining Phase

In defining phase, the problem to be solved or system to be developed is clearly defined. All the requirements to be designed and developed during the development life cycle

Example: Students Examination System Development

Defining the problem

A students Examination System is needed to be developed that covers all the aspects from Examination taking to the Students results generations.

2. PLANNING PHASE

During the planning phase, the objective of the project is determined and the requirements to produce the product are considered. An estimate of resources, such as personnel and costs, is prepared, along with a concept for the new product. All of the information is analyzed to see if there is an alternative solution to create a new product. If there is no other viable alternative, the information is assembled into a project plan and presented to management for approval.

Example

In the **Student Examination System Development** project, planning will be made to set the ultimate goals and an estimate of resources, such as personnel and costs, is prepared.

3. FEASIBILITY

Feasibility study is used to access the strengths and weaknesses of a proposed software/system and present directions of activities, which will improve a project and achieve desired results. The nature and components of feasibility studies depend primarily on the areas in which analyzed projects are implemented.

Feasibility study is the analysis and evaluation of a proposed project/system, to determine, whether it is technically, financially/economically, legally and operationally feasible within the estimated cost and time. Feasibility study is one of the important steps in SDLC.

Types/Forms of Feasibility Study

Feasibility study is divided into the following types/forms

- i. Technical feasibility
- ii. Economic feasibility
- iii. Operational feasibility
- iv. Legal feasibility
- v. Schedule feasibility

Example

The **Students Examination System Development** project is assessed for all the types of feasibilities and presented to management for final approval.

4. Analysis Phase

During the analysis phase, the project team determines the end-user requirements. Often, this is done with the assistance of client focus groups, which provide an explanation of their needs and what their expectations are for the new system and how it will perform.

In this phase, the in-charge of the project team must decide whether the project should go ahead with the available resources or not. Analysis is also looking at the existing system to see what and how it is doing its job.

Questions During Analysis

The project team asks the following questions during the analysis

- i. Can the proposed software system be developed with the available resources and budget?
- ii. Will this system significantly improve the organization

- iii. Does the existing system even need to be replaced etc?

Example

The **Students Examination System Development** project is analyzed for development. The project team will visit the School/College to study the existing and will suggest the possible improvements

5. Requirement Engineering

It is the process of determining user expectations for a new or modified system/software. Requirements engineering is a set of activities used to identify and communicate the purpose of a software system and the framework in which it will be used.

Steps of Requirement Engineering

Requirement engineering consists of the following steps

- i. Requirement gathering
- ii. Requirement validation
- iii. Requirement management

i. Requirement Gathering

Requirement gathering is usually the first part of any software/system development process. In this step meetings with the customers are arranged the market requirements and features that are in demand are analyzed

Types of Requirement Gathering

These requirements are of two types

- a. Functional Requirements
 - b. Non Functional Requirements
- a. **Functional requirements**

Functional requirements specify the software functionality that the developers must build into the product to enable users to accomplish their tasks

b. Non functional requirements

Non functional requirements specify criteria for the judgment of the operations of a system. It describes that how well the system performs its duties.

ii. Requirement Validation

Requirement validation is concerned with examining the requirement to certify that they meet the intentions of the stakeholders. The validation differs from verification in the sense that verification occurs after requirements have been accepted. In requirements validation, the requirements elicited are reviewed to check that requirements are complete and accurate.

iii. Requirements Management

Requirements management is performed to ensure that the software continues to meet the expectations of the acquirer and users.

Requirements management needs to gather new requirements that arise from changing expectations, new regulation or other source of change.

Example

In the students Examination System Development project, the development team will gather the necessary information by interviewing the users. Giving questionnaire or by reading existing documents

6. Design Phase

The design phase is the architectural phase of system design. The flow of data processing is developed into charts, and the project team determines the most logical design and structure for data flow and

storage. For the user interface the project team designs mock up screen layouts that the developers use to write the code for the actual interface

Structure of Design Phase

The design phase normally consists of two different structures. These are

- a. Algorithm
- b. Flowchart

a. Algorithm

An algorithm is a specific step-by-step procedure for carrying out the solution of a problem

Example

In the Students Examination System Development project the following algorithm will find the result of a student on percentage marks

1. Start
2. Read Marks
3. If Marks ≥ 40 THEN Print 'Pass' Else Print 'Fail'
4. End

b. Flowcharts

A flowchart is a type of diagrams that represents an algorithm or a process. It shows the steps of the algorithms with the help of boxes and their order by arrows connecting the boxes. The diagrammatic representation of the algorithm gives a step by step solution to a given problem. The operations are represented in these boxes and the flow of control on the arrows. These flowcharts are used by analyzing designing documenting or managing a process or program in various fields.

Flowchart Symbols

Some most commonly used flowchart symbols are shown in Figure 2.2

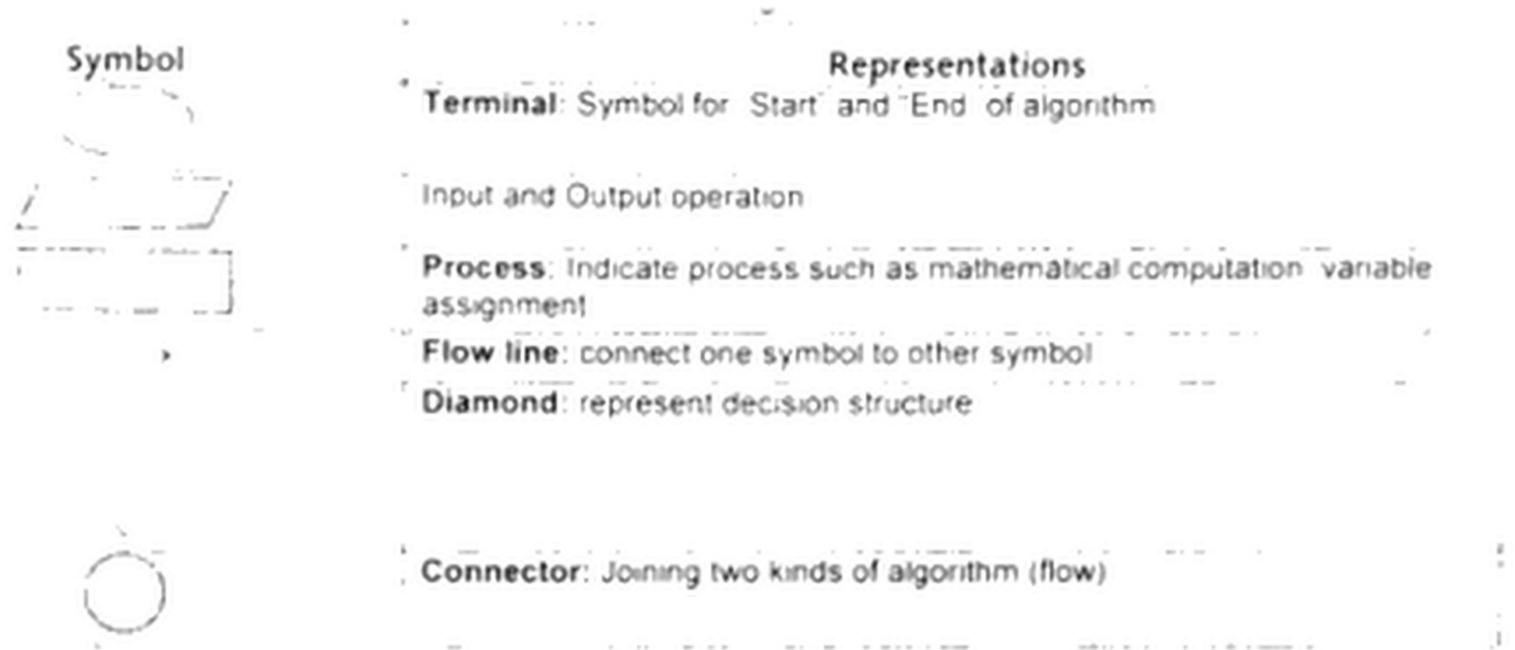


Figure 2.2 Flowchart Symbols

Example

In the Students Examination System Development project, the flowchart for the above algorithm will be as follows

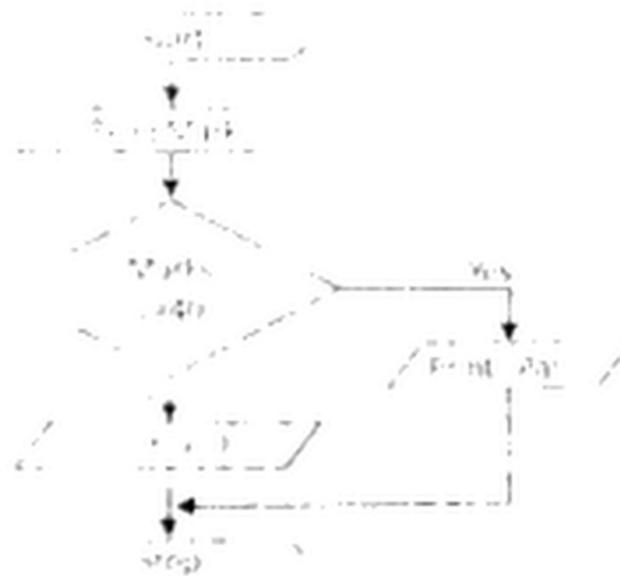


Figure 2.3 Flowchart

7. Construction/Coding

During the construction phase developers execute the plans laid out in the design phase. The developers design the database generate the code for the data flow process and design the actual user interface screens. During the constructions phase, test data is prepared and processed as many times as necessary to refine the code. This code is written in programming languages. Coding is also called computer programming.

Example

In the Student Examination System Development project, the coding style of the C++ programming language is shown as follows

```
/* program to display results of the students */  
#include <iostream.h>  
#include <conio.h>  
Int main ()  
{  
Float Percentage ;  
Cout << "Enter percentage of the student" <<endl;  
Cin>> percentage;  
If (Percentage >=40.0)  
Cout <<'Pass'  
Else  
Cout<<'Fail';  
Getch();
```

8. TESTING/VERIFICATION

During the test phase, all aspects of the system are tested for functionality and performance

Testing

The execution of programming modules to find errors is called testing. Here, the bugs are identified in the programmed modules

Purpose of Testing

The purpose of testing is to evaluate an attribute or capability of a program or system and determine that whether it meets its required results.

Testing/verification the software is actually operating the software under controlled conditions. It is the process of checking the items for consistency by evaluating the results against pre-specified requirements

Example

In the students Examination System Development project the above programming module is tested for errors. The result of the program module is given as follows

```
Test 1:  
Enter Percentage of the student  
70.0  
Pass  
Test 2:  
Enter Percentage of the student  
20.0  
Fail
```

9. Deployment/ Implementation

Software deployment is a set of activities that are used to make the software/system available for use. The deployment is also called implementation.

Main activities involved during Deployment/Implementation Phase

The main activities that are involved during deployment/implementation phase are

- i. Installation and activation of the hardware and software
- ii. In some cases, the users and the computer operation personals are trained on the developed software system

i. Conversion

The process of changing from the old system to the new one is called conversion.

Deployment/Implementation Methods

Following are the four methods/techniques, which are used for system deployment or implementation

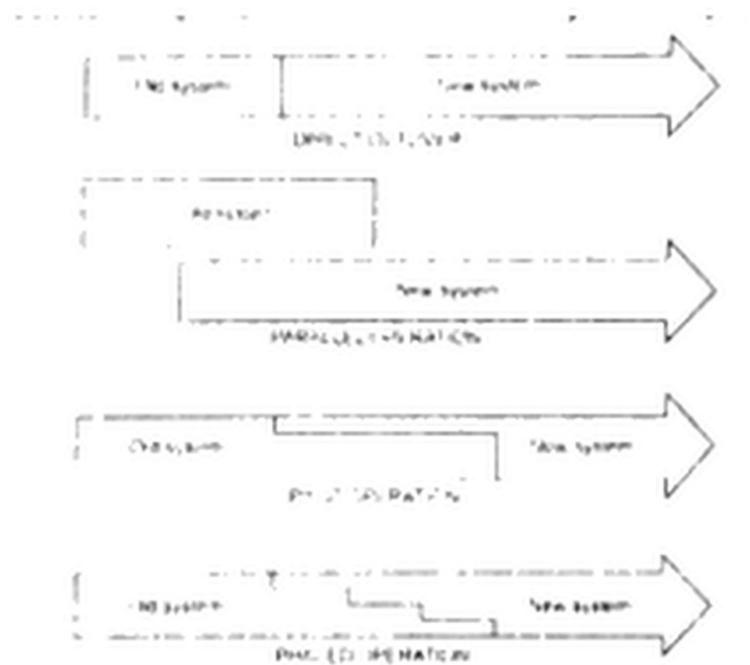


Figure 2.4 Deployment/Implementation

i. **Direct Implementation**

This method involves the old system being completely dropped and the new system being completely implemented at the same time. The old system is no longer available.

ii. **Parallel**

The parallel method of implementation involves operating both systems together for a period. This allows any major problems with the new system to be encountered and corrected without the loss of data

iii. **Phased**

The phased method of implementing from an old system to a new system. The old system is progressively discarded

iv. **Pilot**

With the pilot method of implementation, the new system is installed for a small number of users. These users learn, use and evaluate the new system. Once the new system is deemed to be performing satisfactory then the system is installed and used by all.

Continuous Process

In SDLC, the system maintenance is an ongoing process. The system is monitored continually for performance in accordance with user requirements and needed system modifications are incorporated. When modifications are identified the system may re-enter the planning phase. This process continues until a complete solution is provided to the customer. Maintenance can be either be repairing or modification or some enhancement in the existing system.

2.1.6 PERSONNEL INVOLVED IN SDLC AND THEIR ROLE

Q8. Describe the personnel, which are involved in SDLC. Also state their roles.

Answer

Personnel

Different groups of people and individuals called personnel perform SDLC activities. These personnel are professionals in performing their particular jobs

Types of Personnel

In SDLC, following types of personnel are involved

- i. Management Personnel
- ii. Project Manager
- iii. System Analyst
- iv. Programmer
- v. Software Tester
- vi. Customer

1. Management Personnel/Team

A strong management team has the ability to satisfy the customers and acquires of the software system. Also, a proposed project or a product will only meet its objectives if managed properly, otherwise, will result in failure.

Roles of a Good Management Team

The roles of a good management team are to

- i. Provide consistency of success of the software with regard to time and quality objectives
- ii. Ensure that customer expectations are met
- iii. Collect historical information and data for future use
- iv. Provide a method of thought for ensuring all requirements are addressed through a comprehensive work definition process
- v. **Reduce risks associated with the project**

2. Project Manager

A project manager is a professional responsible for planning, execution and closing of any project. Apart from management skills, a software project manager will typically have an extensive background in software development. She/he is also expected to be familiar with the whole software development life cycle process.

Key roles of a Project Manager

The key roles of a project manager are

- i. Managing the project budget
- ii. Managing the project stakeholders
- iii. Managing the project team
- iv. Managing the project risk
- v. Managing the project schedule
- vi. Managing the project conflicts

3. System Analyst

A system analyst is a professional in the field of software development that studies the problems plans solutions for them recommends software systems and coordinated development to meet business or other requirements. System analyst has expertise in a variety of programming languages, operating systems and computer hardware platforms

Roles & Responsibilities of an Analyst

The general roles and responsibilities of an analyst are defined below

- i. Plan a system flow
- ii. Interact with customers to learn and document requirements that are used to produce business requirement documents
- iii. Define technical requirements
- iv. Interact with designers to understand software limitations
- v. Help programmers during system development phase
- vi. Manage system testing
- vii. Document requirement and contribute to user manuals

4. Programmer

A programmer is a technical persons that writes computer programs in computer programming languages to develop software. A programmer writes tests debugs and maintains the detailed instructions that are executed by the computer to perform their functions.

Responsibilities of a Programmer

The responsibilities of a programmer include

- i. Writing, testing and maintain the instructions of computer programs
- ii. Updating modifying and expanding existing programs
- iii. Testing the codes by running to ensure its correctness
- iv. Preparing graphs tables and analytical data displays which show the progress of a computer program

5. Software tester

A software tester is a computer programmer having specialty in testing the computer programs using different testing techniques

Responsibilities of a software Tester

Software tester is responsible for understanding requirements creating test scenarios test scripts preparing test data executing test scripts, reporting defects and reporting results

6. Customer

A customer is an individual or an organization that is current or potential buyer or user of the software product. Customers usually purchase software manufacturer companies (software houses) users groups and individuals. Customers are the real evaluators of a software product by using it and identifying its merits and demerits.

Clients

Customers are also called clients but the only different between the two is that the customers purchase the software products and the clients purchase services

KEY POINTS

The **Systems Development Life Cycle (SDLC)** is the process of creating or altering information systems and the models and methodologies use to develop these systems

The term **System** is a Greek word meaning to place together. It can be defined as a set of interrelated components having a clearly defined boundary that work together to achieve a common set of objectives

The **basic purpose of System Development Life Cycle** is to develop a system in a systematic way in the perfect manner

A system Development Life Cycle (SDLC) has **three primary objectives** ensure that high quality systems are delivered provide strong management controls over the projects and maximize the productivity

Stakeholders of SDLC are those entities or groups which are either within the organization or outside of the organization that sponsored plan, develop or use a project.

The **planning phase** of SDLC are those entities or groups which are either within the organization or outside of the organization that sponsor plan, develop or use a project

Feasibility study, in SDLC is used to assess the strengths and weakness of a proposed project and present direction of activities, which will improve a project and achieve desired results

During the **analysis phase** of SDLC the project team determines the end user requirements

In SDLC, **requirement engineering** also called requirements analysis is the process of determining user expectations for a new or modified system/software

An **algorithm** is a specified set of instructions for carrying out a procedure or solving a problem

During the **construction phase** of SDLC developers execute the plans laid out in the design phase

The **test phase** of SDLC is used to test the functionality and performance of the system/program

Software deployment in SDLC is a set of activities that are used to make the software system available for use

In SDLC keeping a system in its proper working condition is called **maintenance**

Management is the organization coordination and controlling the activities of a software deployment by the managers and executives in accordance with certain standard procedures

A project manager is a professional responsible for planning, execution and closing of any project.

A **programmer** is a technical person that writes computer programs in computer programming languages to develop software

A **software tester** is a computer programmer having specialty in testing the computer programs using different testing techniques

A **customer** is an individual or an organization that is current or potential buyer or user of the software product.

