



**Daffodil**  
*International*  
**University**

# Lecture 2

## System Development Life Cycles (SDLC)

### Part 1

Course Code: CIS 111

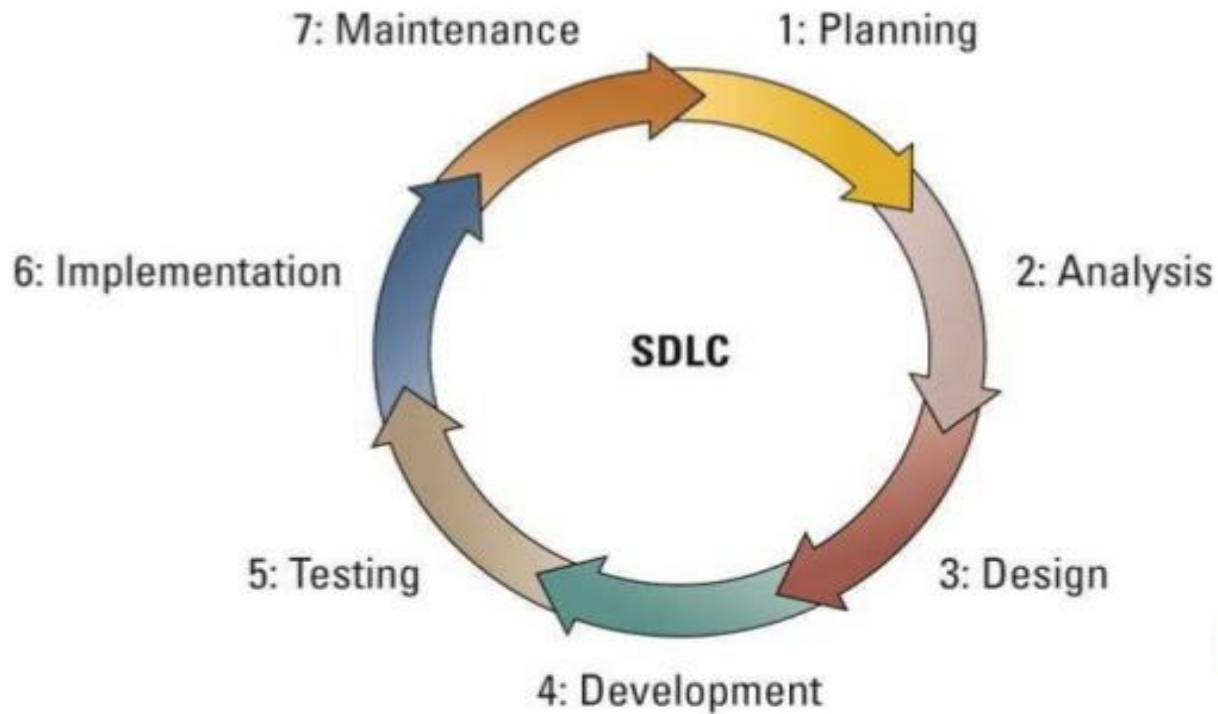
Course Title: Information Systems  
Engineering

# What is SDLC?

- A system development life cycle (SDLC) is the series of identifiable stages that a software product undergoes during its lifetime
- SDLC Models emphasize on the need to follow some structured approach towards building new or improved system.

# Major 7 SDLC Phases:

## SDLC PHASES



# Planning-1<sup>st</sup> Phase

- This is the first phase in the systems development process. It identifies whether or not there is the need for a new system to achieve a business strategic objectives.
- This is a preliminary plan (or a feasibility study) for a company's business initiative to acquire the resources to build on an infrastructure to modify or improve a service.
- The purpose of this step is to find out the scope of the problem and determine solutions. Resources, costs, time, benefits and other items should be considered at this stage.

# Analysis-2<sup>nd</sup> Phase

- After preliminary investigation, analysis phase begins.
- Analysis is a detailed study of
  - the various operations performed by the system,
  - relationships among the various sub systems or functional units
  - and finally the relationships outside the system.
- Many tools are used during analysis. Data flow diagrams, on-site observations, and questionnaires are some examples.
- Detailed work of analysis is kept as a record in a Specification Document.

# Design-3<sup>rd</sup> Phase

- After the system has been analyzed by the analyst, the design stage of system life cycle begins.
- In design phase, the structure or design for the proposed system is finalized.
- Structure of files, databases, input, output, processes, and screens(interfaces) are decided.
- After design is finalized, it is clearly documented in what is called Design Document.

# Development-4<sup>th</sup> Phase

- In this phase, the actual development of the system takes place.
- That is, design representations are translated into actual programs.
- Software developers may install (or modify and then install) purchased software or they may write new, custom-designed programs.
- Programmers are also responsible for documenting the program, providing an explanation of how procedures are coded.

# Testing-5<sup>th</sup> Phase

- After a system has been developed, it is very important to check if it fulfills the customer requirements. For this purpose, testing of the system is done.
- Testing may be repeated, specifically to check for errors, bugs and interoperability. This testing will be performed until the end user finds it acceptable.
- There are various types of tests which are used to test the system. These include unit, integration, and acceptance testing.

# Implementation-6<sup>th</sup> Phase

- Implementation of system means putting up system on user's site.
- This phase involves the actual installation of the newly-developed system. It puts the project into production by moving the data and components from the old system and placing them in the new system via a direct cutover.
- While this can be a risky (and complicated) move, the cutover typically happens during off-peak hours, thus minimizing the risk.

# Maintenance- 7<sup>th</sup> Phase

- Like any system, there is an aging process. Therefore, the system requires periodic maintenance.
- Maintenance can be for software or hardware. User priorities, changes in organizational requirements, or environmental factors call for system enhancements.
- This is very crucial for the system's life.

# In shorter form the key points:

## SDLC Phases & Major Activities

SDLC Phase	Activities
1. Planning	<ul style="list-style-type: none"><li>• Define the system to be developed</li><li>• Set the project scope</li><li>• Develop the project plan including tasks, resources, and timeframes</li></ul>
2. Analysis	<ul style="list-style-type: none"><li>• Gather the business requirements for the system</li></ul>
3. Design	<ul style="list-style-type: none"><li>• Design the technical architecture required to support the system</li><li>• Design system models</li></ul>
4. Development	<ul style="list-style-type: none"><li>• Build the technical architecture</li><li>• Build the database and programs</li></ul>
5. Testing	<ul style="list-style-type: none"><li>• Write the test conditions</li><li>• Perform the testing of the system</li></ul>
6. Implementation	<ul style="list-style-type: none"><li>• Write detailed user documentation</li><li>• Provide training for the system users</li></ul>
7. Maintenance	<ul style="list-style-type: none"><li>• Build a help desk to support the system users</li><li>• Provide an environment to support system changes</li></ul>