# Software Development Lifecycle & Approaches

Topic # 2

---

# What is System Analysis and Design?

- **What is a system?**

- **An organized way of dealing with a problem**

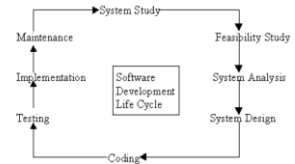- **Analysis and Design of the system?**

---

# System

- A collection of components that work together to realize some objective forms a system.
- Basically there are three major components in every system namely input, processing and output.
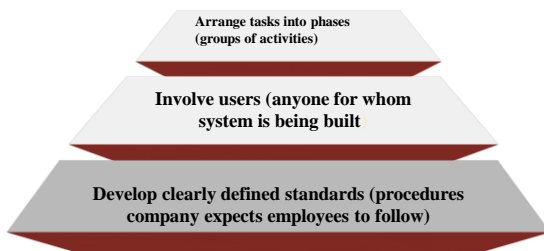
---

# SDLC

- Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality software.
- It consists of a detailed plan describing how to develop and maintain software.
- SDLC consists of many activities/ phases.
- Following are the major phases of SDLC.
  – System study
  – Feasibility study
  – System analysis
  – System design
  – Coding
  – Testing
  – Built release
  – Maintenance

---

# The System Development Life Cycle

What are guidelines for system development?

Arrange tasks into phases (groups of activities)

Involve users (anyone for whom system is being built.

Develop clearly defined standards (procedures company expects employees to follow)

---

# The Players of the System – **Stakeholders**

- **Stakeholder:** Any person, group or organization who is directly or indirectly affected or is involved in the information system.
- 1. **System owners:** An information system's sponsor and executive advocate, usually responsible for funding the project of developing, operating, and maintaining the information system.
  – Interest in s/w system??

## Stakeholders

2. **System User:** A "customer" who will use or is affected by an information system on a regular basis – capturing, validating, entering, responding to, storing, and exchanging data and information.
   – Internal users
   – External users
   – Direct v/s indirect users

## Stakeholders

- **3.System Designer:** A technical specialist who translates system users' business requirements and constraints into technical solution. She or he designs the computer databases, inputs, outputs, screens, networks, and software that will meet the system users' requirements.
- **4.System Builder/developer/coder:** A technical specialist who constructs information systems and components based on the design specifications generated by the system designers.
- **5.Software Quality Assurance People**

## Stakeholders

- **6.Project Manager:** An experienced professional who accepts responsibility for planning, monitoring, and controlling projects with respect to schedule, budget, deliverables, customer satisfaction, technical standards, and system quality.
- **7.External Service Provider (ESP)/consultants:** A systems analyst, system designer, or system builder who sells his or her expertise and experience to other businesses to help those businesses purchase, develop, or integrate their information systems solutions; may be affiliated with a consulting or services organization.
- **8. Other specialists**

## Stakeholders

- **9. System Analyst:** A specialist who studies the problems and needs of an organization to determine how people, data, processes, and information technology can best accomplish improvements for the business.

## The System Development Life Cycle

Who participates in the system development life cycle?
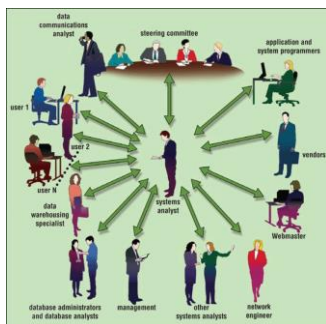


## The System Development Life Cycle

What is the project team?

Formed to work on project from beginning to end

Consists of users, systems analyst, and other IT professionals

Project leader/ manager—one member of the team who manages and controls project budget and schedule

## Stage 1: System Study

- Gives clear picture of what actually the physical system is.
- System study phases(I & II):
  - I: initial survey of the system(scope identification)
  - II: depth study of the system (requirement identification, limitation & issues of the current system). It also includes the back ground analysis and inference or findings of the system.
- Output: system proposal or recommendations to overcome the limitations / issues of the current system.
- Which stakeholders will be involved in this phase?

## Stage 2: Feasibility Study

- A feasibility study precedes the decision to begin a project. It is an assessment of the practicality of a proposed system.
- Three main types of feasibility study:
  - **Technical feasibility:** This assessment is based on an outline design of system requirements, to determine whether the company has the technical expertise to handle completion of the project.
  - **Operational feasibility:** refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project.
  - **Economical feasibility:** A project is considered economically feasible when the benefits that will accrue to the broad community are greater than the cost of undertaking the project.
  - **Schedule Feasibility: Time to complete the project**
- A feasibility study leads to a decision: go or no-go.
- Output: FSR

## What is feasibility?
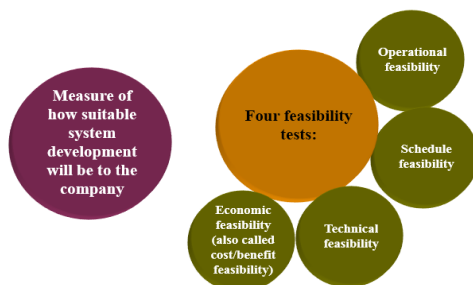


## Stage 3: System Analysis

- Forms the basis of agreement between user and developer. System analysis establish the system's services, constraints and goals by consultation with users.
- It is the study of specifications, operations and relationships with in the system and outside the system.
- Specifies what not how. (Hard task)
- Define the boundary of the new system keeping in view the problems and the new requirement.
- Output: is the Software Requirements Specification (SRS) document.

## Stage 4: System Design

- A major step in moving from *problem* to *solution*.
- Based on system analysis, the new system must be designed.
- *Two main tasks*
  - *General design:* (preliminary design) components and connectors that should be there in the system
  - *Detailed design:* (Detailed Design) logic of modules

**Output:** SDS(system Design Specification)

## Tools and Techniques for Designing

1. Flow Chart
2. Data Flow Diagram
3. Data Dictionary
4. Structured English
5. Decision Tables
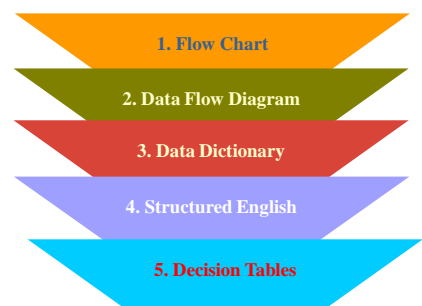
## Example- Decision Table

- **Example: An insurance s/w calculates premium based on the following rules:**
  - **If the insured is male, then premium is $1500 per year**
  - **If the insured is female, then premium is $1000 per year**
  - **If the insured's car has alarm, premium is reduced by 10%**

| Condition/Rule | Rule 1 | Rule 2 | Rule 3 | Rule 4 | Rule 5 | Rule 6 | Rule 7 | Rule 8 |
|---|---|---|---|---|---|---|---|---|
| Male | T | T | T | T | F | F | F | F |
| Female | T | T | F | F | T | T | F | F |
| Car has alarm | T | F | T | F | T | F | T | F |
| Result | X | X | $1350 | $1500 | $1000 | $900 | X | X |

Condition Stubs / Action Stubs

*Rules*

## Stage 5: Coding

- Converts design into **code** in specific language
- **Goal:** Implement the design with simple and easy to understand code
- programs must be modular in nature. This helps in fast development, maintenance and future changes, if required.

- Coding phase affects both **testing** and **maintenance**.
  - Well written code reduces testing and maintenance effort.

- **Output:** is source-code.

## Stage 6: Testing

- **Defects** are introduced in each phase
  - Must be found and removed to achieve high quality
- Software testing is a process of analyzing software for the purpose of finding bugs.
- Using test data, following test runs are carried out
  - Unit test: performed by the respective developers on the individual units of source code to ensure that the individual parts are correct in terms of requirements and functionality.
  - System test: done after unit test. System testing tests the system as a whole. Actual output of the system is matched with the expected outputs. Errors are identified and fixed.
  - User acceptance testing (UAT) – determines if the system satisfies the business requirements

- **Outputs:** are
  - Test plans/results
  - Final tested (reliable) code

## Stage 7: Built Release

- After UAT, deployment phase begins.
- Final phase of SDLC, puts the product into production
- All programs of system are loaded onto the user's computer.
- Then training of user starts including
  - how to execute the package
  - how to enter the data
  - how to process data

## Built Release Strategies

- Parallel run: computerized & manual systems are executed in parallel.
- Advantages of Parallel run:
  - Manual results comparison with the computerized one.
  - Failure of the computerized system at the early stage, does not affect the working of the organization.
- Pilot run: New system is installed in parts. Some part of the new system is installed first and executed successfully for considerable time period.
- Advantages:
  - When results are found satisfactory then only other parts are implemented.
  - This strategy builds the confidence and the errors are traced easily.

## Conversion Plans

- **Abrupt Cutover**
- **Parallel Conversion**
- **Staged / Phased conversion**
- **Location Conversion**

**Pilot Conversion**
The point where old sys completely shifts to new sys is Called total cutover

## Stage 8: Maintenance

- Maintenance phase focuses on **changes** that associated with
  - Error Correction
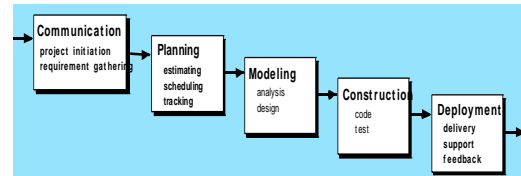  - Platform Adaptations required
  - Enhancement due to change
  - Re-engineering
- Maintenance is required to:
  - eliminate errors in the system during its working life
  - tune the system to any variations in its working environment.
- System Review: is necessary from time to time for:
  - knowing the full capabilities of the system
  - knowing the required changes or the additional requirements
  - studying the performance
- Major change during the review:
  - If a major change to a system is needed, a new project may have to be set up to carry out the change.
  - New project will then proceed through all above life cycle phases.

## Systems Maintenance

- Maintenance
  - Corrective - fix errors
  - Adaptive - satisfy changing needs
  - Perfective - enhance performance
  - Preventative - fix potential problems

## System Environments

- Development
- Test
- Staging
- Pre-Production
- Production
- Mirror
- Roles involved:
- D, PM, T, BA, QA, Designer.

## Alternative S/W Devpt Approaches

- The methodology discussed earlier was common for most projects but however today a variety of types of projects, technologies and development strategies exist – one size no longer fits all.
- For instance building the information system in-house or buying a commercial software solution from a software vendor.



## Alternative routes & strategies

- **Perspective methodologies** follow all rules whereas **adaptive methodologies** change themselves as needed within certain guidelines.
- **Model driven methodologies** develops software using the notion of drawing pictures of the system whereas **product driven methodologies** focus on building the system and see how the users react.
- **Product driven** approaches tend to emphasize either **rapid prototyping** OR writing codes as soon as possible (e.g. Extreme Programming - XP)

## Plan driven versus change driven

- Plan driven
  - Waterfall
  - Incremental/Iterative
- Change driven
  - Agile Methodologies

## The Waterfall Model



32

## The Waterfall Model

- Most widely used; suitable when prob requirements well understood
- Suitable when req well defined and stable
- Suitable when changing existing system
- Leads to blocking state

## The Waterfall Model

- Failure:
  - Real projects rarely follow sequential approach
  - Doesn't support uncertainty/changes in projects .. Nowadays proj are subject to never ending stream of changes
  - Working product only available at the end .. Customer patience
  - The user is involved only in the beginning of the process, while gathering requirements, and at the end, during acceptance testing

## The Incremental Model
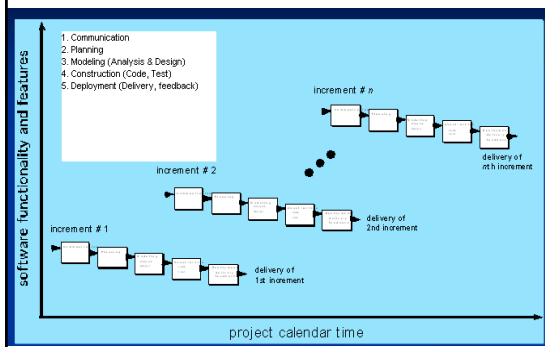


# Agile Methodologies

- Effective (rapid and adaptive) response to change
- Effective communication among all stakeholders
- Drawing the customer onto the team
- Organizing a team so that it is in control of the work performed

*Yielding ...*

- Rapid, incremental delivery of software
- Is driven by customer descriptions of what is required (scenarios)
- Recognizes that plans are short-lived
- Develops software iteratively with a heavy emphasis on construction activities
- Delivers multiple 'software increments'
- Adapts as changes occur

## Cross Life Cycle Activities

**Cross life-cycle activity** – any activity that overlaps many or all phases of the systems development process. Also called **Umbrella Activities.**
- **Fact-finding**
  - **Fact-finding** - the formal process of using research, interviews, meetings, questionnaires, sampling, and other techniques to collect information about system problems, requirements and preferences.
- **Documentation and presentation**
  - **Documentation** – the ongoing activity of recording facts and specifications for a systems for current and future reference.
  - **Presentation** – the ongoing activity of communicating findings, recommendations, and documentation for review by interested users and mangers.
  - **Repository** – a database and/or file directory where system developers store all documentation, knowledge, and artifacts for one or more information systems or projects.
- **Feasibility analysis**
- **Process and project management**

## END OF TOPIC 2

-**COMING UP!!!!!!**
-**SAD v/s OOAD**
-**Requirement Engineering**
-**UML & Use cases**

38