

# **TOPIC 2: INFORMATION SYSTEMS DEVELOPMENT**

**@BBA II 2026**

# Topic Outline

- Information Systems Concept
- Stakeholders in information systems development
- Steps of the information systems development Life cycle. SDLC
- System development methodologies.
- Advantages, Challenges and application of each Methodology.
- End user approaches to system development.
- Activities involved in the implementation of new information systems

# Information System Development

An information system is a set of interrelated components working together to collect, process, store, and disseminate information to support decision making.

Examples of information systems include Academic Management Information System (ACMIS), Accounting systems, Human Resource systems, pay roll information systems, etc.

# What is an Information System

An Information System (IS) can be defined technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization.

In addition to supporting decision making, coordination, and control, IS may also help managers and workers to analyze problems, visualize complex



# Components of an Information System

- Data
- Software
- Hardware
- Telecommunications
- Procedure
- People

# What is System Development?

**“System Development”** refers to the process of examining a business situation, with the intent of improving it through better procedures and methods.

**System Development** has two major components in it

- **System Analysis** – Analysis of current system for Problems & demerits in it and additional requirements in new system
- **System Design** – Process of planning new system which will replace the old one

# Beginning of Systems Development

***System Development*** usually begins when a problem or opportunity is identified by Managers, it can be because of any of the followings:

- New design idea to smoothen the process in the organization.
- Evolving environmental changes such as Competition.
- Adding new business or product line to present business.
- Present system does not satisfy the users information needs.
- Present system no longer efficiently and effectively meets the Goals of organization.
- Excessive time spent in correcting errors.
- Current Reports / Outputs not meeting users decision making skills.
- Escalating customer and vendor complaints.

# Options for Obtaining Information Systems

1. Build your own
2. Buy a prepackaged system from a software development company or consulting firm. Example: Payroll system.
3. Outsource development to a 3<sup>rd</sup> party: outside organization custom build a system to an organization's specifications. Good option when an organization does not have adequate resources or expertise.
4. End user Computing/ development: Individual users and departments build their own custom systems to support their individuals. Example MS. Excel.

# Examples of Information Systems Commonly Used in Government of Uganda

1. Integrated Personnel & Payroll Management System (IPPS)
2. Integrated Financial Management Information Systems (IFMIS)
3. Computerized Lands Management System (CLMIS)
4. Health Management Information System (HMIS)
5. Local Government Management Information system (LGMIS)
6. Academic Information Management System (ACMIS)
7. Academic Registration Information System (ARIS)
8. Teacher Management Information System (TMIS)
9. Electronic Court Case Management Information System (ECCMIS) (Judiciary Uganda)

# Examples of Information systems commonly used in Uganda's Ministries and Agencies

- Electronic Health information systems used in Hospitals, Pharmacies and drug stores
- Academic Management Information System (ACMIS)
- Intelligent Transport Management Information System (ITMIS)
- Human Capital Management Information System (HCMIS)
- Education Management Information System (EMIS)
- Integrated Loan Management Information System (ILMIS)
- EFRIS stands for the Electronic Fiscal Receipting and Invoicing Solution. It is a real-time digital system implemented by the Uganda Revenue Authority (URA) to manage and track the issuance of invoices and receipts by taxpayers, helping to reduce revenue leakage and widen the tax base.

# Parish Based Management Information Systems (PBMIS)

Ministry of Local Government is the over all lead agency for the PBMIS and is responsible for reporting on Parish Development Model (PDM).

The Parish Chiefs will be the primary data collectors under the PBMIS, and will coordinate the community groups, and NGOS/CSOs to provide operational data relevant to PDM

# Parish Based Management Information Systems (PBMIS)

Parish Based Management Information System is among the 7 Pillars of the Parish Based Model of the Government of Republic of Uganda.

The Parish Based Management Information System (PBMIS) is an Integrated system that seeks to support community profiling, data collection, analysis, tabulation, storage and dissemination at the Parish level.

This will support planning and decision making at the Grassroots and also at National Level

# **Examples of Information systems commonly used in Uganda's businesses**

- Enterprise Resource planning (ERP) systems
- Customer Relationship Management (CRM) systems
- Inventory Management Systems such as SAP
- Point of sale (POS) systems
- Accounting software such as QuickBooks, Tally, Navision etc
- Human Resource Management Systems
- Supply Chain Management Systems
- E-commerce platforms such as Jumia
- Electronic Health information systems
- Agricultural information systems etc.

# Stakeholders in Information System development

- Discuss the different stakeholders and their roles in system development
  - ✓ Users
  - ✓ Managers
  - ✓ Sponsors
  - ✓ System analysts
  - ✓ System designers
  - ✓ System developers Etc.

# Participants in Systems Development

- **Stakeholders**

- Individuals who ultimately benefit from the systems development project, vendors and suppliers

- **Users**

- Individuals who interact with the system regularly

- **Owners /Sponsors/Top Management**

- Financers of the Project

- **Systems analyst and designer**

- A professional who specializes in analyzing and designing business systems

# Participants in Systems Development

- **Programmer**
  - The individual responsible for modifying or developing programs to satisfy user requirements
- **Technical Specialist**
  - Experts in certain types of technology e.g. Telecom, databases, who help create I.S.
- **Vendors**
  - They provide/sell software and hardware to the organisation.

# System End Users

These are the people who use computers to perform their jobs, like desktop operators.

End users can be divided into various categories.

- ☐ Hands-on users.
- ☐ Indirect end users
- ☐ Senior managers

# The Users of a System (System End Users)

- **Hands-on users.** They actually interact with the system. They are the people who feed in the input data and get output data.
- **Indirect end users** who do not interact with the systems hardware and software. However, these users benefit from the results of these systems. These types of users can be managers of organization using that system.

# The Users of a System (System End Users)

- Those who have management responsibilities for application systems. These oversee investment in the development or use of the system.
- **Senior managers.** They are responsible for evaluating organization's exposure to risk from the systems failure.

# The Systems Analyst

- A systems analyst, also known as business technology analyst, is an information technology (IT) professional who specializes in analyzing, designing and implementing information systems based on the business/operational needs of the organization. He or she has both business and technical knowledge
- A **systems analyst** is responsible for studying, analyzing, and designing information systems to meet the specific needs of an organization.

# Roles of Systems Analyst

- Studying, understanding and documenting current systems.
- Gathering information and determining requirements.
- The analyst usually undertakes a series of interviews, communicating to users, IT Managers, systems specialists etc
- Recommending systems solutions.

# The Systems Analyst



# Skills of Systems Analyst

- Business knowledge.
- Interpersonal skills.
- Problem solving skills.
- Working knowledge of IT.
- Computer programming knowledge.
- Communication skills.
- Listening skills
- Flexibility and Adaptability to various systems.

# System Development Life cycle (SDLC)

- Any information systems project follows a logical series of development phases. These are known as the systems development lifecycle.
- In the beginning there were no defined stages that people would follow in information system development
  - The situation was chaotic. Systems built:-
    - would take long to be build.
    - were expensive.
    - would not meet all users requirements.
    - would contain a lot of errors.

# Why Initiate new System Development?

- Problem with existing system
- Desire to exploit new opportunities
- Increasing competition
- Desire to make effective use of information
- Organization growth
- Merger and acquisition
- Increased customer complaints
- Change in external environment eg regulations
- New processes and product lines

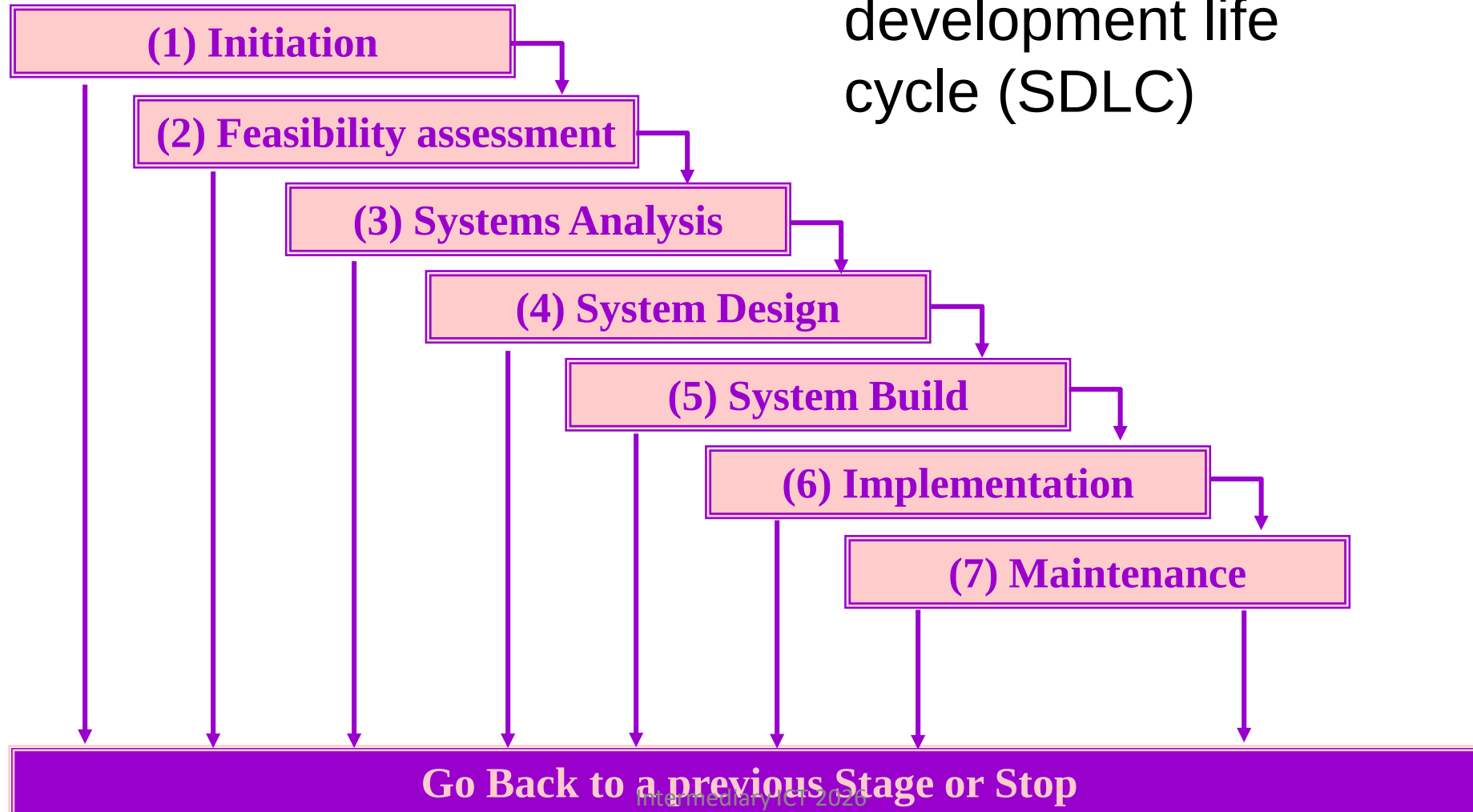
# System Development Life cycle (SDLC)

## **SDLC stages:**

- **Initiation/Preliminary Investigation**
- **Feasibility assessment**
- **System analysis**
- **Systems design**
- **System build and Testing**
- **Implementation**
- **Maintenance**

# SDLC

A 7-stage systems development life cycle (SDLC)



# SDLC (1) Preliminary Investigation/Initiation

The startup phase in an IS development project. Its aims are to recognize a **problem** or **opportunity** and try to **brainstorm on the possible**

The objective of this activity is **to review all requests and identify those proposals that are most beneficial** to the organization.

This is done through a feasibility study in areas of economic, technical, legal, operational and feasibility.

The output of the Investigation phase is ***Preliminary investigation report***

## 2. Feasibility assessment

- ❑ An activity at the start of the project to ensure that the project is a viable business proposition.
- ❑ Trying to find out if the selected project is feasible and then prepare to ensure that the project is successful.
- ❑ The feasibility report analyses the need for and impact of the system and considers different alternatives for acquiring software.
- ❑ Feasibility of the system is divided into :-
  - ❑ Technical feasibility
  - ❑ Economic Feasibility
  - ❑ Operational feasibility
  - ❑ Time Feasibility

**Output:** feasibility report and recommendation to proceed.

# (Feasibility Assessment Parameters) Continuation

- **Economical Feasibility:** Assessment of whether a project is an acceptable financial risk and if the organization can afford the expense needed to complete it.
- **Technical Feasibility:** Assessment of whether hardware, software and communications components can be developed or acquired to solve a business problem.
- **Operational Feasibility:** Finding views of workers, employees, customers & suppliers about the use of new system.
- **Schedule Feasibility:** Estimation of time to take new system to become operational.
- **Legal feasibility:** Testing whether new system will comply financial reporting requirements & Company's contractual obligations.

# SDLC (3) Systems Analysis

- ***Analysis phase*** - involves end users and IT specialists working together to gather, understand, and document the business requirements for the proposed system.
- It involves a **detailed study of the weaknesses & problems in the existing system** so as to gather requirements for the new or improved system.
- Requirements under study for the new system include business functional and non-functional requirements.
- The output of this stage of system development is the ***System Requirements document***

# SDLC (4) Systems Design

- *Design phase* - build a technical blueprint of how the proposed system will work
- Deliverable is the *Technical design* that specifies:
  - System outputs, inputs, user interfaces;
  - Hardware, software, databases, telecommunications, personnel & procedures;
  - Blueprint of how these components are integrated.
- Logical system design states *what* the system will do, using abstract specifications such as data and their structures.
- Physical system design states *how* the system will perform its functions, with actual physical specifications

# SDLC (5) System Development/Coding

- ***Development phase*** – this involves transforming of the design blue prints into an **actual working system** (Computer code)
- Different programming languages and development software are used to develop turn designs into systems e.g. HTML, CSS, Java etc.
- Software developers play a great role at this stage of system development.

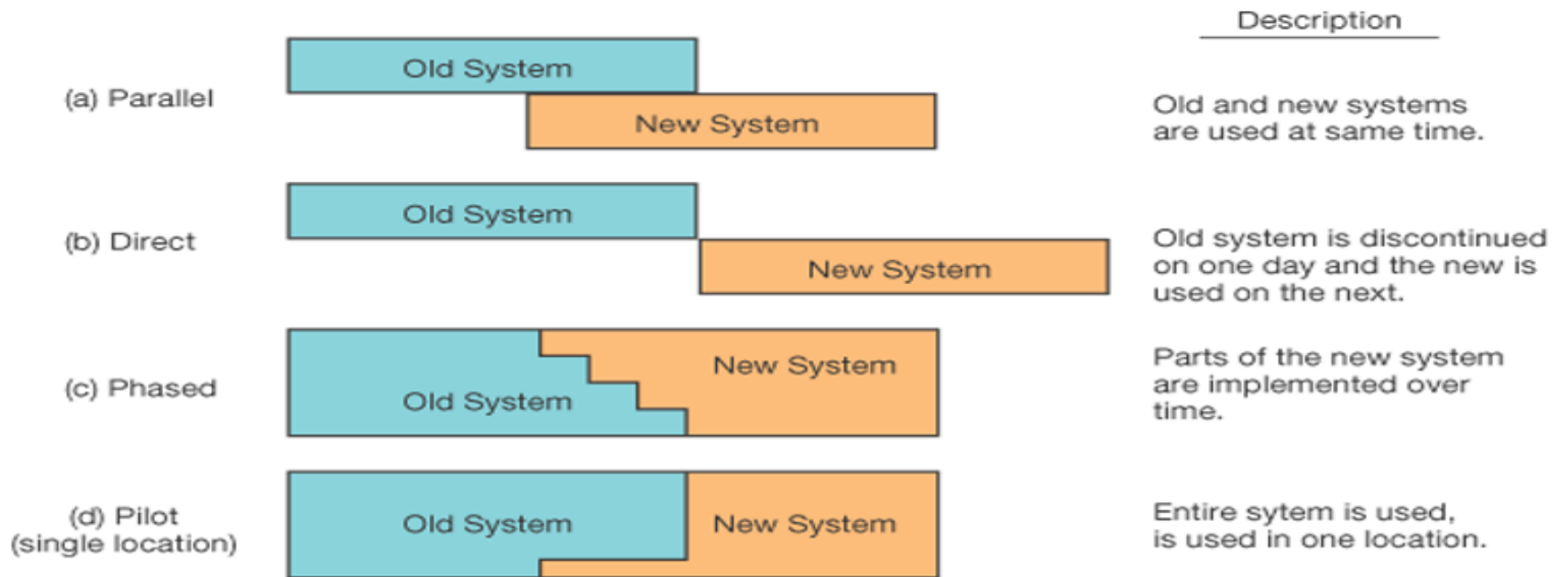
# SDLC (5) System Testing

- This verifies that the system works and meets all of the requirements defined in the analysis phase. Different system testing can be used
  - ***Unit testing*** – tests individual units of code
  - ***System testing*** – verifies that the units of code function correctly when integrated
  - ***Integration testing*** – verifies that separate systems work together
  - ***User acceptance testing (UAT)*** – determines if the system satisfies the business requirements
  - ***Alpha***: Software tester tests whether it meets design specifications
  - ***Beta***: Actual system users test the capability of the system in the user environment with actual data

# SDLC (6) System Implementation

- **Implementation phase** – is the process of converting from the old system to the new system. Organizations use four major conversion strategies ; Parallel , Direct , Phased and Pilot

**Figure 8.19** Software conversion strategies.



# System Implementation Continuation

- **Parallel implementation** – use both the old and new system simultaneously
- **Plunge (Direct) implementation** – discard the old system completely and use the new
- **Pilot implementation** – start with small groups of people on the new system and gradually add more users
- **Phased implementation** – implement the new system in phases
- Another activity that takes place at this stage is training of the users as well preparing relevant documentations to help in using of the new system.

# SDLC (7) System Maintenance

- ***Maintenance phase*** - monitor and support the new system to ensure it continues to meet the business goals
- Different types of maintenance can be carried out i.e.
  - **Corrective Maintenance:** Reactive modification of the system by resolving problems/error/faults/bugs discovered in the system
  - **Perfective maintenance:** Modification of a software product after delivery to improve performance

# System Maintenance Continuation

- **Preventive maintenance:** Modification of a software product after delivery to detect and correct latent or hidden faults in the software product before they become effective faults. This type of maintenance may be one of the most cost effective, since if performed timely and properly, it can avoid major problems with the system. E.g. installing an anti-virus
- **Adaptive maintenance:** Modification of a software product performed after delivery to keep a software product usable in a changed or changing business environment.

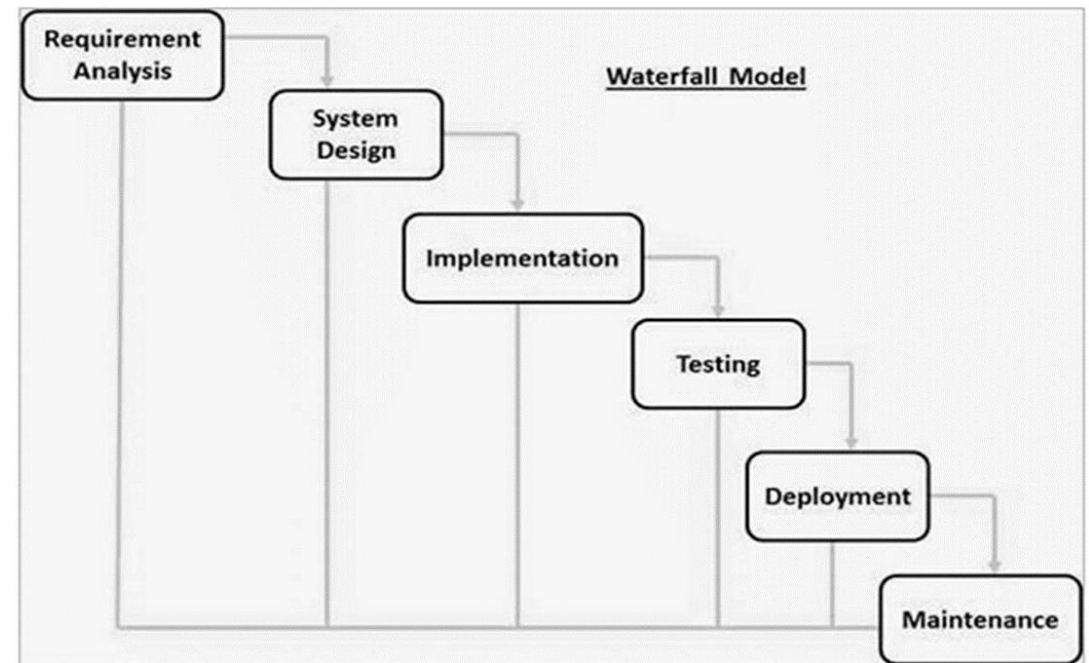
# Information System Development Methodologies

A system development methodology refers to the framework or method used to structure, plan, and control the process of developing an information system. Various methodologies can be used to include:

- The waterfall model
- Prototype model
- The spiral model
- Rapid Application Development
- Joint Application Development
- Agile Development approach etc.

# Waterfall Model

- In this model, analysts and users **proceed in sequence from one stage of system development to another.**
- Output from one phase is the input for the next phase
- The Waterfall Model is the earliest method of structured system development.
- Although it has come under attack in recent years for being too rigid and unrealistic when it comes to quickly meeting customer's needs, the Waterfall Model is still widely used.



# Strengths of Waterfall Model

- Ideal for supporting less experienced project teams and project managers, or project teams whose composition fluctuates.
- The orderly sequence of development steps and strict controls for ensuring the adequacy of documentation and design reviews helps **ensure the quality, reliability, and maintainability** of the developed software.
- Minimizes changes to the system requirements due to a long time taken analyzing them.
- Its simple and easy to understand and use

# Weaknesses of Waterfall Model

- Inflexible, slow, costly and cumbersome due to significant structure and tight controls.
- One stage of system development should be complete before another one starts. **Delays, change in trends**
- Depends upon early identification and specification of requirements, yet **users may not be able to clearly define what they need early** in the project.
- Problems are often not discovered until system testing.
- Takes a lot of time associated with lengthy deliverables.

# PROTOTYPING

- ***Prototyping*** - the process of building a model that demonstrates the features of a proposed product, service, or system
- ***Prototype*** - a model of a proposed product, service, or system

# Prototype Model

- In this methodology, a quick and dirty program (prototype) that demonstrate the features of a proposed system is developed and users work with it.
- Users provide feedback to the developers so that they refine it. This continues until the users are satisfied.

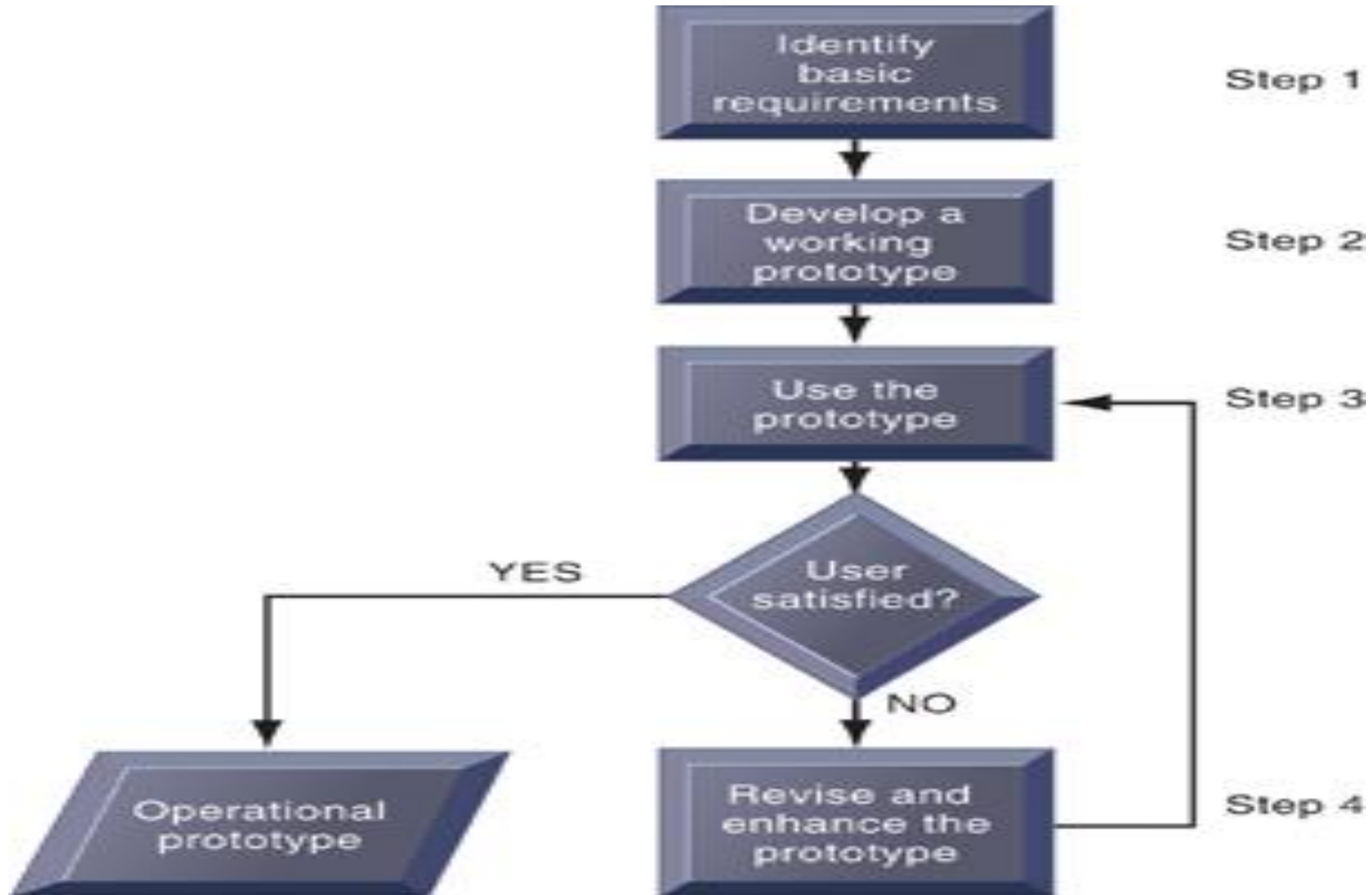
# STEPS IN PROTOTYPING

- **Step 1: Identify the user's basic requirements.** The system designer (usually an information systems specialist) works with the user only long enough to capture the user's basic information needs.
- **Step 2: Develop an initial prototype.** The system designer creates a working prototype quickly, using tools for rapidly generating software.

# STEPS IN PROTOTYPING Con't

- **Step 3: Use the prototype.** The user is encouraged to work with the system to determine how well the prototype meets his or her needs and to make suggestions for improving the prototype
- **Step 4: Revise and enhance the prototype.** The system builder notes all changes the user requests and refines the prototype accordingly. After the prototype has been revised, the cycle returns to step 3. Steps 3 and 4 are repeated until the user is satisfied.

# The Prototyping Process



# Situations where prototyping is most appropriate

- Project is large with many users, interrelationships, and functions, where project risk relating to requirements definition needs to be reduced
- Project objectives are unclear.
- User is not fully knowledgeable.
- Pressure exists for immediate implementation of something

# Advantages and Disadvantages of Prototyping

## Advantages

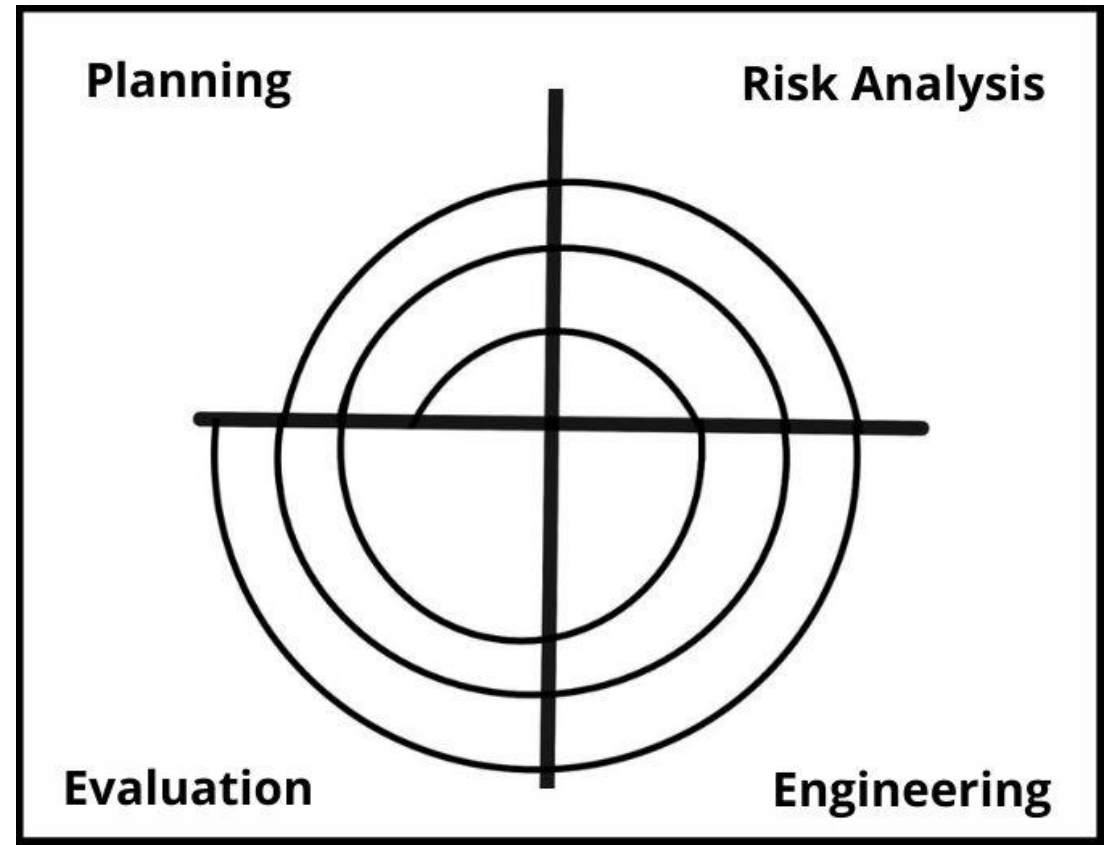
- Encourages active knowledge worker participation
- Helps resolve discrepancies among knowledge workers
- Gives knowledge workers a feel for the final system
- Helps determine technical feasibility
- Helps sell the idea of a proposed system

## Disadvantages

- Leads people to believe the final system will follow shortly
- Gives no indication of performance under operational conditions
- Leads the project team to forego proper testing and documentation

# Spiral Model

- Spiral model is an evolutionary software process model which is a combination of an iterative nature of prototyping and systematic aspect of the traditional waterfall model
- Provides support for risk handling



# Situations where spiral model is appropriate

- When costs and risk evaluation is important
- For medium to high risk projects
- Users are unsure of their need
- Requirements are unclear and complex
- Significant changes are expected

# Spiral Model

## Advantages

- Changing requirements can be accommodated
- Allows for extensive use of prototypes
- Requirements can be captured more accurately
- Users see the system early
- Early and frequent feedback from users

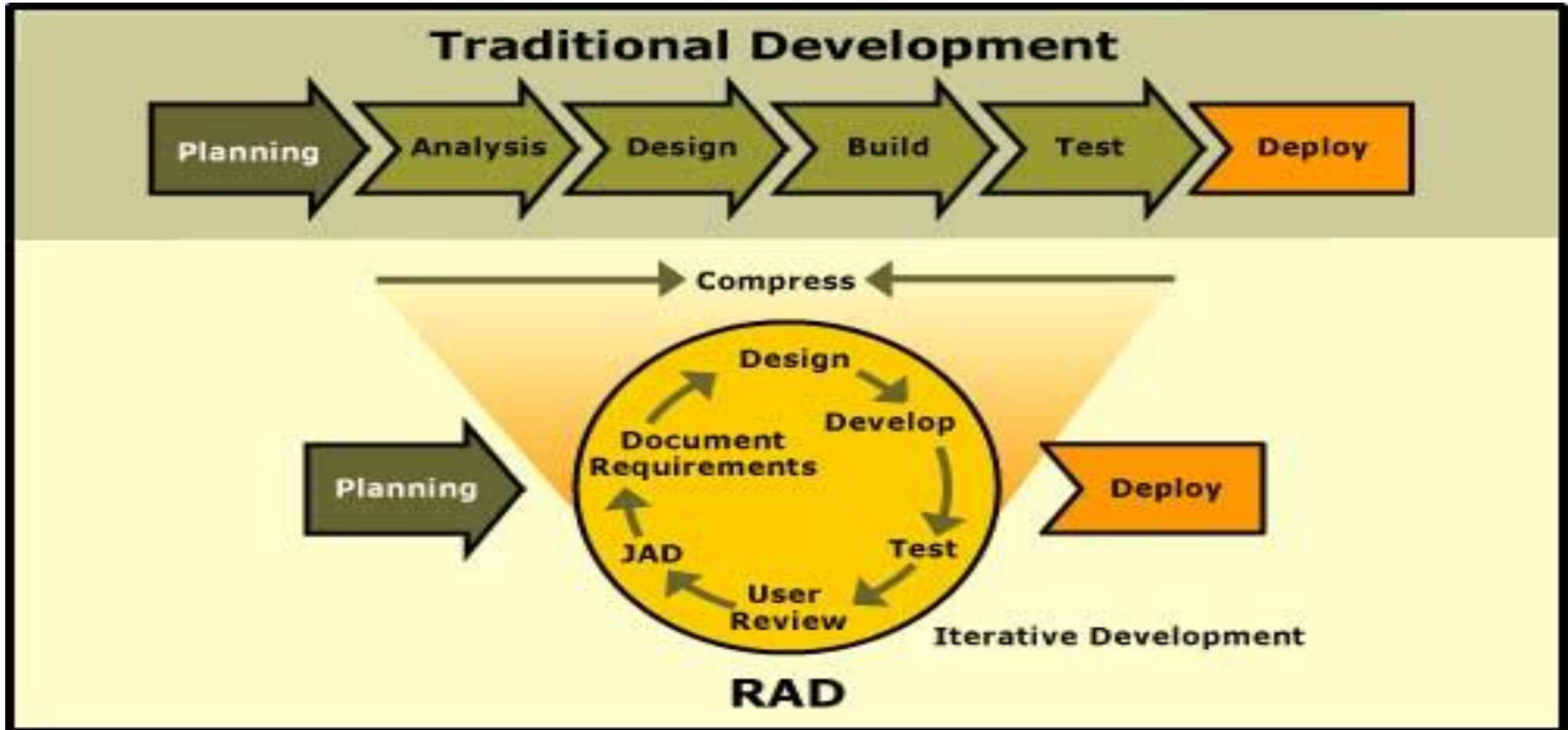
## Disadvantages

- End of project may not be known early
- Not suitable for small or low risk projects since it could be expensive for small projects
- Process is complex
- Risk assessment expertise is required

# Rapid Application Development (RAD)

- A technique that employs tools, techniques, and methodologies designed to speed application development.
- It compresses the analysis, design, build, and test phases into a series of short, iterative development cycles. This makes it possible to develop systems faster and get them into the hands of users to work with.
- Staffed with small integrated teams of developers, end users, and IT technical resources for speed, unity of vision and purpose, effective informal communication and simple project management

# Illustration



# When to Use RAD

- When the team includes programmers and analysts who are experienced with it and are available
- When there are pressing reasons for speeding up application development
- When Users are sophisticated and highly engaged with the goals of the company
- If resources are available and there is need to produce the system in a short span of 2-3 months

# Rapid Application Development (RAD)

## Advantages

- Tighter fit between user requirements and system specifications
- Works especially well where speed of development is important
- Ability to rapidly change system design as demanded by users
- Strong user stake and ownership of system
- Concentrates on essential system elements from user viewpoint

## Disadvantages

- May try and hurry the project too much
- Loosely documented
- May not address pressing business problems
- Potentially steep learning curve for programmers inexperienced with RAD tools
- More speed and lower cost may lead to lower overall system quality
- Difficulty with module reuse for future systems

# Joint Application Development (JAD)

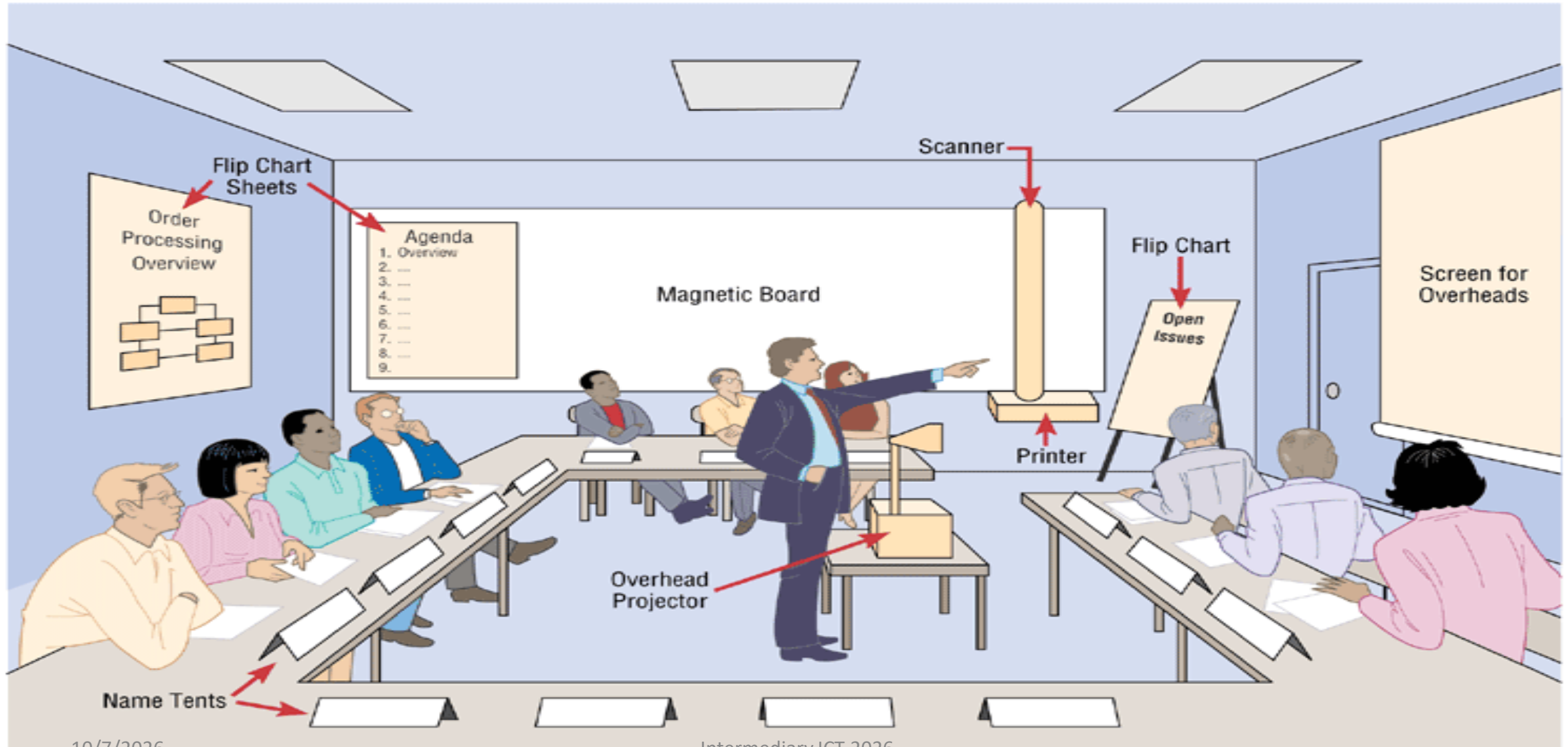
- *Joint Application Development (JAD)* involves coming together of different stakeholders and working together for several days in a series of intensive meetings so as to review system requirements.
- JAD is most often used within the systems analysis and systems design stages of the SDLC.
- In contrast to the SDLC requirements analysis, JAD has a group meeting in which all users meet simultaneously with analysts.
  - Used to accelerate generation of information requirements and to develop initial systems design
  - Brings end users and information systems specialists together in interactive session to discuss system's design
  - Can significantly speed up design phase and involve users at intense level

# JAD

JAD participants typically include:

- ✓ Facilitator – facilitates discussions, enforces rules
- ✓ End users – 3 to 5, attend all sessions
- ✓ Developers – 2 or 3, question for clarity
- ✓ Observers – 2 or 3, do not speak
- ✓ Managers- 2 or 3, Attend all sessions
- ✓ Subject Matter Experts – limited number for understanding business & technology

# Con't



# Joint Application Development (JAD)

## Advantages

- It allows effective participation among team members
- Better understanding of project objectives
- Shorter project completion times
- Commitment to the success of project deliverables
- Every one speaks out their minds

## Disadvantages

- Different opinions within the team makes it difficult to align goals and maintain focus
- Depending on the size of the project, JAD may require a significant time commitment
- Meetings of this nature are not frequent
- Costly to run

# End-user development/End User Computing

- Uses **fourth-generation languages** to allow end-users to develop systems with little or no help from technical specialists
- **Fourth generation languages:** Less procedural than conventional programming languages
- The development and support of IT systems by knowledge workers with little or no help from IT specialists

# Advantages & Disadvantages of End User System Development

## Advantages

- Improves requirements determination
- Increases knowledge worker participation and sense of ownership
- Increases speed of systems development

## Disadvantages

- Inadequate knowledge worker expertise leads to inadequately developed systems
- Lack of organizational focus creates “privatized” IT systems
- Insufficient analysis of design alternatives leads to subpar IT systems
- Lack of documentation and external support leads to short-lived systems

# Outsourcing

- ***Outsourcing*** - the delegation of specific work to a third party for a specified length of time, at a specified cost, and at a specified level of service
- ***Offshore outsourcing*** - using organizations from other countries to write code and develop systems
- IT outsourcing takes on 1 of 4 forms:
  - Purchasing existing software
  - Purchasing existing software and pay the publisher to make certain modifications
  - Purchasing existing software and pay the publisher for the right to make modifications yourself
  - Outsourcing the development of an entirely new and unique system for which no software exists

# Advantages and Disadvantages of Outsourcing

## Advantages

- Focus on unique core competencies
- Exploit the intellect of another organization
- Better predict future costs
- Acquire leading-edge technology
- Reduce costs
- Improve performance accountability

## Disadvantages

- Reduces technical know-how for future innovation
- Reduces degree of control
- Increases vulnerability of strategic information
- Increases dependency on other organizations

# Criteria for selecting a Methodology

- **Clarity of User requirements:** when user requirements are not clear, prototyping and RAD are appropriate.
- **Familiarity with technology:** when the developers are not familiar with the technology that they need to develop a system, a lot of risks are involved, therefore methodologies such as Spiral model are preferred.
- **System complexity:** some systems are more complex to develop, therefore choice should be of in-depth analysis. Spiral, Waterfall are appropriate etc

# Criteria for Selecting Methodology Continues...

- **System Reliability:** Some systems e.g. for medical equipment, missile control systems etc need a high degree of reliability, comprehensive testing should be done during the cycle, e.g. in XP and V-Model
- **Short time schedules:** systems that have a short time scope for development need methodologies like RAD and prototyping. Waterfall method is the most time consuming.
- **Schedule visibility:** some systems need constant checking to determine whether its on schedule. There is need to measure pace of development; RAD is the most appropriate.

# Criteria for Selecting Methodology

- **Need for end user involvement:** systems that require high participation of end users would dictate on the methodology to use. The best to use would be JAD, Prototyping, RAD

# Problems when Initiating New or Modified Systems

- Fear that employee will lose his/ her job, power, or influence within the organization
- Belief that the proposed system will create more work than it eliminates
- Reluctance to work with “computer people”/IT technicians
- Anxiety that proposed system will negatively alter organization’s structure
- Belief that other problems are more pressing, or that the system is being developed by people unfamiliar with “the way things need to get done”
- Unwillingness to learn new procedures or approaches

# Assignment-Submit after one 1week

1. As an officer taking part in systems development, carefully study your business processes in your organization and state any 5 user requirements that you feel should be included in the new system to be developed for your organization
2. What is Agile Methodology?, how does it work?
3. Discuss challenges that organizations encounter during the development of information systems, highlighting the critical factors that influence the success or failure of these projects. and analyze the strategies organizations can employ to overcome these challenges and ensure the effective implementation of information systems.

**The Assignment should be hand written**

