Topic 2
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

In today's rapidly evolving digital landscape, information systems development plays a fundamental role in shaping the efficiency and effectiveness of organizations, whether its designing user friendly applications or implementing robust databases.

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 accounting systems, human resource systems, pay roll information systems etc.

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

General industry trends and technologies relevant to Information system development

- Digital transformation
- Cloud computing
- Mobile applications
- Cybersecurity
- Data analytics and artificial intelligence
- Blockchain technology
- E-government initiatives
- Fintech innovations
- Improved skills development 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.

Case study 1: A Case Study on Information System Development

Title: Streamlining Student Enrollment

- **Introduction**

In the fast-paced world of higher education, efficient student enrollment processes are crucial for both students and institutions. This case study explores the development of an information system aimed at streamlining the student enrollment process at EleD&A university. The goal is to provide educators and students with insights into the complexities, challenges, and successes associated with information system development.

- **Background**

EleD&A University, a renowned institution, faces numerous **challenges with its manual enrollment process**. These challenges include long wait times, data entry errors, and difficulties in tracking student progress. To address these issues, the university has decided to develop a comprehensive information system to modernize the enrollment process.

- **Problem Statement**

EleD&A University has identified several key issues with the existing enrollment process, High operational costs due to manual data entry and processing, Lengthy wait times for students during enrollment, Inaccurate record-keeping leading to data discrepancies, Limited accessibility to enrollment services for remote and international students.

- **Objectives**

The university aims to develop an information system to achieve the following objectives, Reduce enrollment processing time by 50%, Eliminate data entry errors and discrepancies, Enhance accessibility and usability for both students and staff, Facilitate data analysis for decision-making.

Assignment 1

Provide an IT solution (select an activity or process) for an organization or business of your choice. On $1^{1/2}$ pages clearly highlight a Title, an (1) introduction, (2) background, (3) problem statement and (4) objectives of your area of study.

- **Project title**
- **Introduction**
- **Background**
- **Problem Statement**
- **Objectives of the study**

Examples of Information systems commonly used in Uganda's businesses

- Enterprise Resource planning (ERP) systems
- Customer Relationship Management (CRM) systems
- Inventory Management Systems
- Point of sale (POS) systems
- Accounting software
- Human Resource Management Systems
- Supply Chain Management Systems
- Business Intelligence (BI) and Analytics tools
- E-commerce platforms
- Customized software solutions
- Health information systems
- Agricultural information systems etc.

System Development Life cycle (SDLC)

SDLC is the overall process of developing information systems through a multi-step process from investigation of initial requirements through analysis, design, development, implementation and maintenance

System Development usually begins by identifying a problem or opportunity. This can be because of a new design idea to smoothen the process in the organization, evolving environmental changes such as competition, present system not satisfying the users information needs etc.

SDLC (1) Preliminary Investigation

Whatever may be the reason, a request is submitted from different stake holders to the IS department. A system analyst is assigned to make a preliminary investigation.

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 schedule feasibility.

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

(Preliminary Investigation) 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 (2) 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 (3) 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.
- Physical system design states *how* the system will perform its functions, with actual physical specifications

SDLC (4) System Development

- ***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.

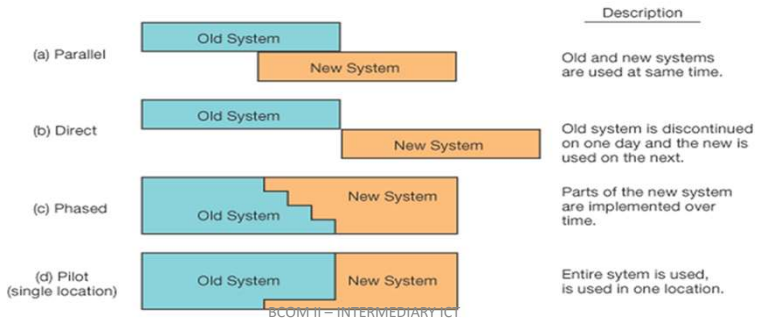
SDLC (5) System Testing

- 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 a software product performed after delivery to correct discovered problems. This usually refers to problems that were not identified during the implementation phase. An example of remedial maintenance is the lack of a user-required feature or the improper functionality of it.
  - **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 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.

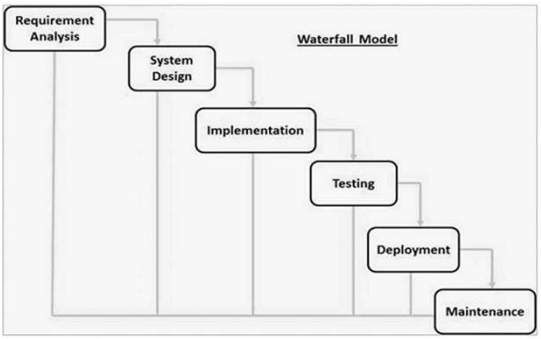
Information System Development Methodologies

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.



Strengths of Waterfall Model

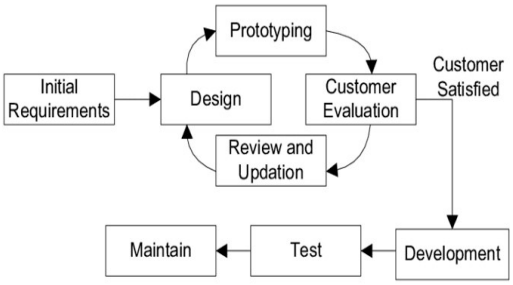
- Ideal for supporting less experienced project teams and project managers, or project teams whose composition fluctuates.
- The orderly sequence of development steps and strict controls for ensuring the adequacy of documentation and design reviews helps ensure the quality, reliability, and maintainability 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.

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.



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

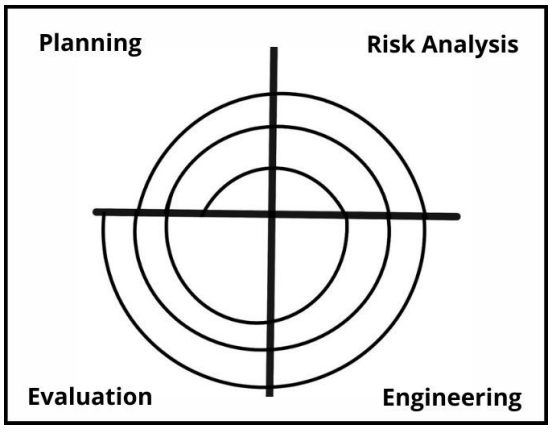
- Encourages active knowledge worker participation
- Helps resolve discrepancies among knowledge workers
- Gives knowledge workers a feel for the final system
- Helps determine technical feasibility
- Helps sell the idea of a proposed system

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

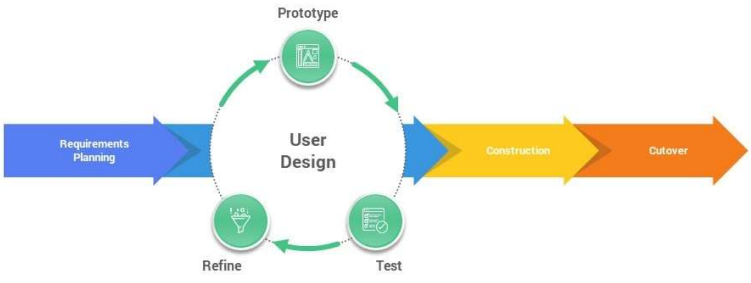
- Changing requirements can be accommodated
- Allows for extensive use of prototypes
- Requirements can be captured more accurately
- Users see the system early
- Early and frequent feedback from users

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)

- *Rapid application development (RAD)* methodology and tools makes it possible to develop systems faster and get them into the hands of users to work with.
- This facilitates a better understanding of the system and suggest revisions that bring the system closer to what is needed by the users.
- Typical RAD tools (CASE tools) used include the following:
  - Graphical User Interface development environment
  - Reusable components
  - Code generator
  - Programming languages



When to Use RAD

- The team includes programmers and analysts who are experienced with it
- There are pressing reasons for speeding up application development
- 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

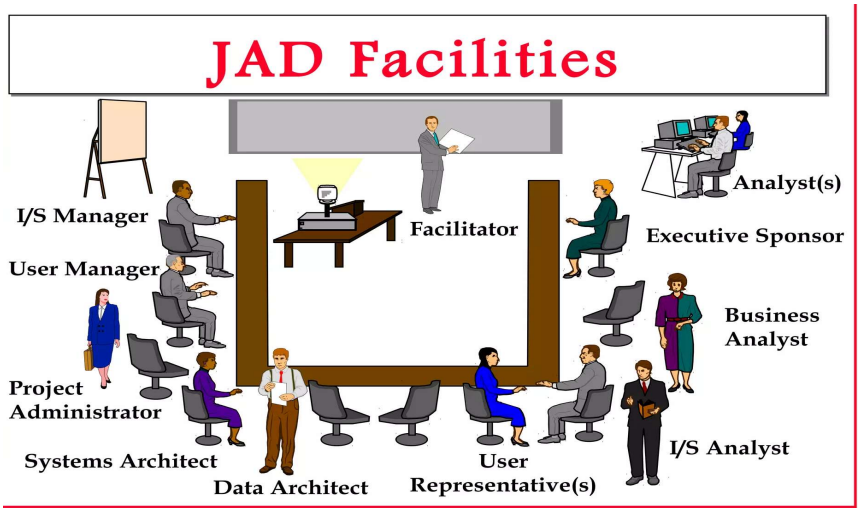
- Tighter fit between user requirements and system specifications
- Works especially well where speed of development is important
- Ability to rapidly change system design as demanded by users
- Strong user stake and ownership of system
- Concentrates on essential system elements from user viewpoint

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

- *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



Joint Application Development (JAD)

Advantages

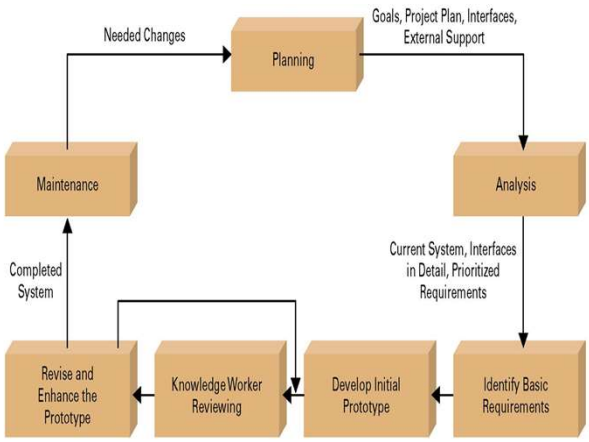
- It allows effective participation among team members
- Better understanding of project objectives
- Shorter project completion times
- Commitment to the success of project deliverables

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

End-user development

- 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

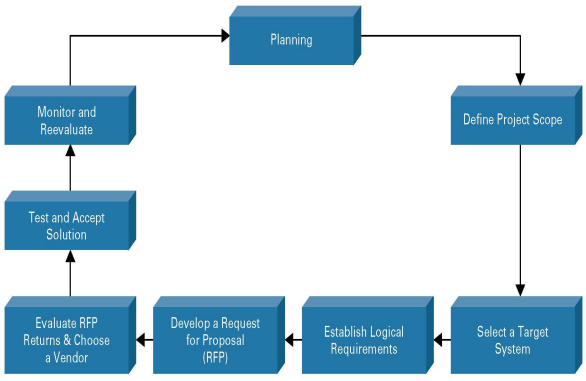
- ***Outsourcing*** - the delegation of specific work to a third party for a specified length of time, at a specified cost, and at a specified level of service
- ***Offshore outsourcing*** - using organizations from other countries to write code and develop systems
- IT outsourcing takes on 1 of 4 forms:
  - 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

When to Use Outsourcing

- When there is only a limited opportunity for the firm to distinguish itself competitively through a particular information systems application or a series of applications
- When outsourcing does not strip the company of the technical know-how required for future information systems innovation
- When the firm's existing information systems capabilities are limited, ineffective, or technically inferior

Figure 6.8

The Outsourcing Process



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 Appropriate Methodology

- The traditional SDLC approach often works well for large projects with well-defined requirements, where there is *not* a lot of time pressure.
- Prototyping requires effective management to make sure that the iterations of prototyping do not continue indefinitely.
- RAD may be less appropriate than conventional programming languages for larger projects, or those with a lot of real-time processing.
- JAD is easy for senior management to understand, yet it is difficult and expensive to get all people in the same place at the same time.
- *End-user development* is also a possibility for larger projects whose priorities are not high enough to lead to a timely response from the central IS unit.
- *Outsourcing* should always be considered by large and complex systems with a significant risk of failure, yet the disadvantages should be reviewed carefully.

Assignment 2

1. How does the Agile development approach differ from traditional waterfall methodology, and what are the key principles and practices that make Agile an effective framework for modern software development projects?
2. 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.