Chapter – 5 Acquisition, Development and Implementation of Information Systems

1. ACHIEVING SYSTEM DEVELOPMENT OBJECTIVES [PROBLEMS]

PM

RTP

1. **User Related Issues**: Issues where user/customer is reckoned as the primary agent
    a. ***Shifting User Needs***: User requirements for IT are constantly changing
    b. ***Resistance to Change***: People have a natural tendency to resist change
    c. ***Lack of User Participation***:
    d. ***Inadequate Testing and User Training***:
2. **Developer Related Issues**:
    a. ***Lack of Standard Project Management and System Development Methodologies***:
    b. ***Overworked or Under-Trained Development Staff***:
3. **Management Related Issues**:
    a. ***Lack of Senior Management Support and Involvement***:
    b. ***Development of Strategic Systems***:
4. **New Technologies**:

(question may be asked as 'why organization fail to achieve SDLC objective)

2. ACCOUNTANTS' INVOLVEMENT IN DEVELOPMENT WORK

An accountant can help in various related aspects during system development; some of them are as follows:

1. **Return on Investment (ROI)**:
    a. ***Cost***: This includes estimates for typical costs involved
        i. Development Costs:
        ii. Operating Costs:
        iii. Intangible Costs:
    b. ***Benefits***:
        i. subdivided into tangible and intangible benefits.
        ii. how the system development effort has benefitted an organization.
2. **Computing Cost of IT Implementation and Cost Benefit Analysis**:
3. **Skills expected from an Accountant**: An accountant must possess skills to understand the system development efforts including understanding of the business objectives, expert book keeper etc.

3. SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)

The System Development Life Cycle provides system designers and developers to follow a sequence of activities. A phase of the SDLC is not complete until the appropriate documentation or artifact is produced which are called deliverables.

4. ADVANTAGES OF SDLC

1. Better planning and control by project managers;
2. Compliance to prescribed standards ensuring better quality;
3. Documentation that SDLC stresses on is an important measure of communication and control; and
4. The phases are important milestones and help the project manager and the user for review and signoff.

© CA KUNAL AGRAWAL

5. ADVANTAGES OF SDLC FROM PERSPECTIVE OF THE IS AUDIT

PM

RTP

1. The **IS auditor can have clear understanding of various phases of the SDLC**
2. The IS Auditor on the basis of his/her examination, **can state in his/her report about the compliance by the IS management** of the procedures, if any, set by the management.
3. The IS Auditor, if has a technical knowledge and ability of different areas of SDLC, can be a **guide during the various phases of SDLC**.
4. The IS auditor can provide an **evaluation of the methods and techniques used** through the various **development phases of the SDLC**.

6. SHORTCOMINGS AND ANTICIPATED RISKS ASSOCIATED WITH THE SDLC

1. The development team may find it cumbersome.
2. The users may find that the end product is not visible for a long time.
3. The rigidity of the approach may prolong the duration of many projects.
4. IT may not be suitable for small and medium sized projects.

7. PHASES OF SDLC

PM

PHASE	PHASE NAME	NATURE OF ACTIVITY
1.	Preliminary Investigation	Determining and evaluating the strategic feasibility of the system and ensure that the solution fits the business strategy.
2.	Systems Requirements Analysis	Analyzing the typical system requirements, in view if its functionalities, deliverables etc.
3.	Systems Design	Designing the system in terms of user interface, data storage and data processing functions on the basis of the requirement phase by developing the system flowcharts, system and data flow diagrams, screens and reports.
4.	Systems Development	Developing the system as per the system designed in view of its adequate implementation to lead to fulfillment of requirements to the satisfaction of all the stakeholders.
5.	Systems Testing	Requisite testing to ensure the valid and reliable implementations.
6.	Systems Implementation	Operationalization of the developed system for the acceptance by management and user before migration of the system to the live environment and data conversion from legacy system to the new system.
7.	Post Implementation Review and Maintenance	Continuous evaluation of the system as it functions in the live environment and its updation / maintenance.

© CA KUNAL AGRAWAL

:: WWW.FACEBOOK.COM/PAGE.KUNAL :: KUNALSIR.IN ::
CONTACT # 9929503735/8233920041

8. PRELIMINARY INVESTIGATION

- ✓ It is aimed to **determine and analyze the strategic benefits** in implementing the system.
- ✓ The deliverable of the preliminary investigation includes a report including feasibility study observations.
- ✓ Following **steps** are involved in PRELIMINARY INVESTIGATION:
  - ❖ Identification of Problem,
  - ❖ Identification of objectives,
  - ❖ Delineation of scope, and
  - ❖ Feasibility Study

Delineation of Scope

PM

- ❖ The scope of a solution defines its **typical boundaries**.
- ❖ It should be clear and comprehensible to the user management stating the extent and **'what will be addressed by the solution and what will not**'.
- ❖ The typical scope determination may be performed on the following dimensions:
  1. ***Functionality Requirements***:
  2. ***Data to be processed***:
  3. ***Control Requirements***:
  4. ***Performance Requirements***:
  5. ***Constraints***:
  6. ***Interfaces***:
  7. ***Reliability requirements***

- ❖ Two primary methods with the help of which the scope of the project can be analyzed are given as follows:

PM

RTP

1. ***Reviewing Internal Documents***: The analysts co**nducting the investigation first try to learn about the organization** involved in, or affected by, the project. For example, **to review an inventory system proposal, an analyst may try to know how does the inventory department operates**. This can be done by studying written operating procedures.
2. ***Conducting Interviews***: **Written documents may not show/describe users' views about current operations.** To learn these details, analysts use interviews. Interviews allow analysts to know more about the nature of the project request and the reasons for submitting it.

Feasibility Study

- ❖ A feasibility study is carried out by the system analysts, which refers to a process of evaluating alternative systems through cost/benefit analysis.

1. **Financial Feasibility**: The solution proposed may be **prohibitively costly**

2. **Economic Feasibility**: It includes an evaluation of all the **incremental costs and benefits expected** if the proposed system is implemented.

PM

The purpose of estimating the following:

a. The cost of conducting a full systems investigation;
b. The cost of hardware and software for the class of applications being considered;
c. The benefits in the form of reduced costs or fewer costly errors; and
d. The cost if nothing changes (i.e. the proposed system is not developed).

© CA KUNAL AGRAWAL

3. **Technical Feasibility**: It is concerned with issues pertaining to hardware and software. The technical issues usually raised during the feasibility stage of investigation include the following:
    a. Does the necessary **technology exist** to do what is suggested and **can it be acquired**?
    b. Does the **proposed equipment** have the technical **capacity** to **hold the data** required to use the new system?
    c. Can the proposed application be **implemented with existing technology**?
    d. Will the proposed system **provide adequate responses to inquiries**, regardless of the number or location of users?
    e. Can the system be **expanded if developed**?
    f. Are there technical guarantees of **accuracy, reliability, ease of access**, and data security?
4. **Schedule or Time Feasibility**:
5. **Resources Feasibility**:
6. **Operational Feasibility**: It is concerned with ascertaining the views of workers, employees, customers and suppliers about the use of computer facility. the operational feasibility of a project are stated below:
    a. Is there **sufficient support for the system from management** and from users?
    b. Are current business methods acceptable to users?
    c. Have the users been involved in planning and development of the project?
    d. Will individual performance be poorer after implementation than before?
7. **Behavioral Feasibility**:
8. **Legal Feasibility**:

9. *SYSTEM REQUIREMENT ANALYSIS*

Objectives: This phase includes a thorough and detailed understanding of the current system, identifies the areas that need modification to solve the problem, the determination of user/managerial requirements and to have fair idea about various systems development tools.

The following activities are performed during this phase:

1. To identify the stake owners.
2. To consult the stake owners to determine their expectations and resolve their conflicts.
3. To analyse requirements to detect and correct conflicts and determine priorities.
4. To verify that the requirements are complete, consistent, unambiguous, verifiable, testable and traceable.
5. To model activities such as developing models to document Data Flow Diagrams, ER Diagrams (Entity relationship diagram) and data dictionary.
6. To document activities such as interview, questionnaires, reports etc.

Document/Deliverable: A Systems Requirements Specification (SRS) report.

a) Fact finding Techniques: Fact-finding techniques used by the system analyst are:

1. **Documents**:
2. **Questionnaires**:
3. **Interviews**:
4. **Observation**:

PM
PM
PM

© CA KUNAL AGRAWAL

b) Analysis of the Present System

PM

The following areas should be studied in depth:

1. ***Review historical aspects***:
2. ***Analyse inputs***:
3. ***Review data files maintained***:
4. ***Review methods, procedures and data communications***:
5. ***Analyze outputs***:
6. ***Review internal controls***:
7. ***Model the existing physical system and logical system***:
8. ***Undertake overall analysis of present system***:

c) System Development Tools

Many tools and techniques have been developed to improve current information systems and to develop new ones. Such tools help end users and systems analysts to –

1. **Conceptualize, clarify, document and communicate the activities and resources involved** in the organization and its information systems
2. **analyze present business operations**
3. Propose and design new or improved information systems

The major tools used for system development can be grouped into four categories based on broader features:

PM

1. ***System components and flows***: These tools help the system analysts to document the data flow. A data flow diagram uses a few simple symbols to illustrate the flow of data among external entities
2. ***User interface***: Designing the interface between end users and the computer system is a major consideration of a system analyst.
3. ***Data attributes and relationships***:
    a. A Data Dictionary catalogues the description of the attributes (characteristics) of all data elements and their relationships.
    b. Entity-Relationship diagrams are used to document the number and type of relationship among the entities in a system.
    c. File layout forms, document the type, size and names of the data elements.
4. ***Detailed system process***: These tools are used to help the programmer develop detailed procedures and processes.

We will now describe some of these tools in detail:

1. ***Structured English***:
   Program Design Language is the use of the English language
2. ***Flowcharts***:
3. ***Data Flow Diagrams***:
   A Data Flow Diagram uses few simple symbols to illustrate the flow of data among external entities (such as people or organizations, etc.),
4. ***Decision Tree***:
5. ***Decision Table***:
    a. ***CASE (Computer-Aided-Software Engineering) Tools***: CASE tool refers to those software which help to automatically develop high quality, defect free and maintainable software.
6. ***System Components matrix***:

© CA KUNAL AGRAWAL

a. A System Component Matrix provides a matrix framework to document the resources used, the activities performed and the information produced by an information system.

7. ***Data Dictionary***:
   a. A data dictionary is a computer file that contains detail information about the data items in the files/database of a business information system.
8. ***Layout form and Screen Generator, Menu Generator, Report generator, Code Generator***:
   Layout form and Screen Generator:
   Menu Generator:
   Report Generator:
   Code Generator:

d) Systems specifications:

At the end of the analysis phase the system analyst prepares a document called "systems requirement specifications (SRS)" which contains the following:

1. ***Introduction***:
2. ***Information Description***:
3. ***Functional Description***:
4. ***Behavioral Description***:
5. ***Validation Criteria***:
6. ***Appendices***:
7. ***SRS Review***:

10. SYSTEMS DESIGN

Objective: Design an Information System that best satisfies the user / managerial requirements.

Key design phase activities include

1. Describing inputs and outputs, such as screen design and reports
2. Determining the processing steps and computation rules for the new solution
3. Determining data file or database system design
4. Preparing the program specifications for the various types of requirements and
5. Internal / external controls.

Document / Deliverable: Creates a 'blueprint' for the design with the necessary specifications for the hardware, software, people and data resources.

The design phase involves following steps:

(a) Architectural Design
(b) Design of the Data / Information Flow
(c) Design of the Database
(d) Design of the User-interface
(e) Physical Design; and
(f) Design of the hardware/system software platform

1. ***Architectural Design***:
   a. major modules
   b. function and scope of each module
   c. interface features of each module
   d. modules that each module can call directly or indirectly and

2. ***Design of Data / Information flow***:

Page 6 © CA KUNAL AGRAWAL

a. existing data / information flows
b. problems with the present system and
c. objective of the new system.

3. ***Design of Database***: Design of the database involves determining its scope i.e. local database V/S global structure. The scope is decided on the basis of interdependence among organizational units.

4. ***Design of User-Interface***:
   a. source documents to capture raw data
   b. hard-copy output reports
   c. screen layouts for source document input
5. ***Physical Design***: During physical design, the primary concern of the auditor is effectiveness and efficiency issues. Some of the issues addressed here are
   a. Type of hardware for client application and server application
   b. Processing – batch – online, real – time
   c. Frequency of input, output
6. ***Design of System Software Platform***:
7. ***Internal Design Controls***:
   a. Whether management reports of stage I and stage II, were referred by System Designer?
   b. Whether all control aspects have been properly covered?

11. SYSTEMS ACQUISITION AND SOFTWARE DEVELOPMENT

SYSTEM ACQUISITION

After a system is designed either partially or fully, the next phase of the systems development starts which relates to the acquisition of hardware, software and services.

1. ***Acquisition Standards***: Management should establish documented acquisition standards that should focus on –

- Ensuring **security, reliability, and functionality** already built into a product.
- Ensuring that managers **review the appropriate vendors, contracts**, and licensing agreements.
- Ensuring that **new products are compatible with existing systems**.
- Ensuring that **functional, security, and operational requirements** are accurately identified and clearly detailed in request-for-proposals (RFP).

2. ***Acquiring Systems Components from Vendors***:

The following considerations are valid for both acquisition of hardware and software:

1. ***Vendor Selection***: Vendor selection is to be done prior to sending RFP.
2. ***Geographical Location of Vendor***: The issue to look for whether the vendor has local support persons.
3. ***Presentation by Selected Vendors***:
4. ***Evaluation of Users Feedback***:

Some specific considerations for hardware and software acquisition are described as follows:

1. The benchmark tests to be done for proposed machine. .
2. The benchmarking problems are oriented towards testing whether a computer offered by the vendor meets the requirements of the job on hand of the buyer.

Page 7 © CA KUNAL AGRAWAL

3. Tests should enable buyer to effectively evaluate cross performance of various systems

3. Other Acquisition Aspects and Practices:

1. ***Hardware Acquisition***:
2. ***Software Acquisition***:
3. ***Contracts, Software Licenses and Copyright Violations***:
4. ***Validation of vendors' proposals***:
5. ***Methods of validating the proposal:***
    a. **Checklists**:
    b. **Point-Scoring Analysis**:
    c. **Public Evaluation Reports**:
    d. **Bench marking problem for vendor's proposals**:
    e. **Test problems**:

12. SYSTEMS DEVELOPMENT: PROGRAMMING TECHNIQUES AND LANGUAGES

Objective: To convert the specification into a functioning system.
Activities: Application programs are written, tested and documented, conduct system testing.
Document / Deliverable: **A fully functional and documented system**.

PM

A good coded program should have the following characteristics:

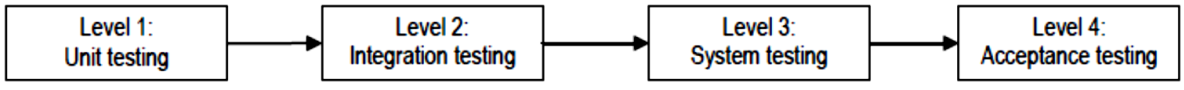
- ✓ ***Reliability***: It refers to the consistence which a program provides over a period of time.
- ✓ ***Robustness***: taking into account all possible inputs
- ✓ ***Accuracy***: It refers not only to what program is supposed to do
- ✓ ***Efficiency:*** performance which should not be unduly affected with the increase in input
- ✓ ***Usability***: It refers to a user-friendly interface and easy-to-understand
- ✓ ***Readability***: It refers to the ease of maintenance of program

Some other issues in systems development:

1) ***Program Coding Standards***:
2) ***Programming Language***:
3) ***Program Debugging***:
4) ***Test the program***:
5) ***Program Documentation***:
6) ***Program Maintenance***:

13. PHASE 5. SYSTEMS TESTING

Different levels of Testing are as follows:



PM

Unit Testing

- ✓ programmer tests if **individual units of source code are fit for use**. A unit is the smallest testable part of an application which may be an individual program, function, procedure, etc.
- ✓ There are five categories of tests that a programmer typically performs on a program unit:

© CA KUNAL AGRAWAL

- ***Functional Tests***:
- ***Performance Tests***:
- ***Stress Tests***:
- ***Structural Tests***:
- ***Parallel Tests***:

✓ ***Types*** of Unit Testing:

- ***Static testing*** (i.e. testing without actual running of program)
  - **Desk Check**:
  - **Structured walk-through**:
  - **Code inspection**:
- ***Dynamic testing*** (i.e. testing by actual running of program)
  - **Black Box Testing**: **The test designer selects valid and invalid inputs and determines the correct output**.
  - **White Box Testing**: White box testing uses an internal view of the system to design test cases based on internal structure. **It requires programming skills to identify all paths through the software**.
  - **Gray Box Testing**: Gray box testing is a software testing technique that uses a combination of black box testing and white box testing.

PM

Integration Testing

Integration testing is an activity of software testing in which **individual software modules are combined and tested** as a group.

1) ***Bottom-up Integration***:
2) ***Top-down Integration***:
3) ***Regression Testing***: Each time a new module is added as part of integration testing, the software changes. In the context of the integration testing, the regression tests ensure that changes or corrections have not introduced new errors.

System Testing

1) ***Recovery testing***: 'how well the application is able to recover from crashes
2) ***Security testing***: Information System protects data and maintains functionality as intended or not.
3) ***Stress testing***: determine the stability of a given system or entity.
4) ***Performance testing***: to determine the speed or effectiveness of a computer

Final Acceptance Testing

1) ***Quality assurance testing***:
2) ***User acceptance testing***:
   1. **Alpha Testing**: often performed by the users within the organization.
   2. **Beta Testing**: generally performed by the external users.

14. SYSTEMS IMPLEMENTATION AND MAINTENANCE

Objective: To implement the new system i.e. put it into production.

Document / Deliverable: A full functional / documented system in its operational environment.

Activities: The activities involved in System Implementation are as follows:

1. Conversion of data to the new system files.
2. Training of end users.
3. Completion of user documentation.
4. System changeover.

© CA KUNAL AGRAWAL

5. Evaluation of the system at regular intervals.

Activities during Implementation Stage:

The activities involved in system implementation stage are as follows:

1. **Equipment Installation**:
   a. ***Site Preparation***:
   b. ***Installation of new hardware / software***:
   c. ***Equipment check out***:
2. **Training Personnel**:
3. **System Implementation Conversion Strategies**:
   a. ***Direct / Abrupt Conversion***: In this conversion scheme the old system is discarded and new system is implemented at the same time.
   b. ***Phased Conversion***: In this conversion scheme the old system is discarded in a phased manner and the new system is also implemented module-by-module.
   c. ***Parallel Conversion***: In this conversion scheme the new system is implemented but the old system also continues to work for some time until the new system becomes completely reliable.
   d. ***Pilot implementation***: Other than the above mentioned conversion strategies there can be "Pilot implementation" also. The new system is first implemented in modules of non-critical units and if it is successful then it is moved to larger unit.
4. **Activities involved in conversion**:
   1) ***Procedure conversion***: Operating procedures should be completely documented for the new system that applies to both computer-operations and functional area operations.
   2) ***File conversion:*** Because large files of information must be converted from one medium to another, this phase should be started long before programming and testing are completed. Adequate control must be there.
   3) ***System conversion***: daily processing can be shifted from the old system to the new system. All transactions initiated after this time are processed on the new system.
   4) ***Scheduling personnel and equipment***: Schedules should be set up by the system manager in conjunction with departmental managers which are using the new system.

15. POST IMPLEMENTATION REVIEW (PIR) AND SYSTEMS MAINTENANCE

PIR should be done some time after the solution has been deployed. Typical periods range from 6 weeks to 6 months, depending on the type of solution and its environment.

There are three basic dimensions of Information system that should be evaluated:

1. ***Development evaluation***:
2. ***Operation evaluation***:
3. ***Information evaluation***:

System Maintenance

1) **Scheduled maintenance**:
2) **Rescue maintenance**: Rescue maintenance refers to **previously undetected malfunctions**
3) **Corrective maintenance**: The need for corrective maintenance is usually initiated by bug reports drawn up by the end users.

PM

4) **Adaptive maintenance**: Adaptive maintenance consists **of adapting software to changes in the environment, such as the hardware or the operating system**.
5) **Perfective maintenance**:
6) **Preventive maintenance**:

16. AUDITORS ROLE IN SDLC

The audit of system under development can have three main objectives

1. To provide an opinion on the efficiency, effectiveness and economy of project.
2. The new system being developed provides for adequate audit trails and controls to ensure the integrity of data processed and stored
3. To assess the controls being provided for the management of the system's operations

The auditor can achieve the above objectives by:

1) Attend project and steering committee meetings
2) Examine project documentation
3) Conducting interview
4) Check the compliance with development standards

Some control considerations for an auditor to examine are:

1) Good infrastructure
2) Trained development staff
3) Appropriate approvals
4) Separation of development and test environment
5) Standards are followed
6) Version control
7) Safety of source code
8) Systems maintains proper audit trail

17. SYSTEMS DEVELOPMENT METHODOLOGY

A system development methodology is a formalized, standardized, documented set of activities used to manage a system development project.

All systems development methodologies are characterized by the following common features:

1) ***Division of project***:
2) ***Every stage should have some deliverables***:
3) ***Signoffs/approvals at every stage***:
4) ***Testing***:
5) ***User Training***:
6) ***Change management***:
7) ***Post implementation review (PIR)***:

18. TRADITIONAL / WATERFALL APPROACH / SEQUENTIAL APPROACH:

Basic Principles or Characteristics:

PM

- Project is divided into sequential phases, with some overlap and splash back
- Emphasis is on planning, time schedules, target dates, budgets and **implementation of an entire system at one time**.
- Tight control is maintained over the life of the project

© CA KUNAL AGRAWAL

Strengths:

1. Ideal for less experienced project teams
2. reviews help ensure the quality, reliability, adequacy and maintainability
3. Progress of system development is measurable.
4. Conserves resources.

PM

Weaknesses

1) It is criticized to be Inflexible, slow, costly, and cumbersome
2) Project progresses forward, with only slight movement backward.
3) There is a little to iterate, which may be essential in some situations.
4) Problems are often not discovered until system testing.

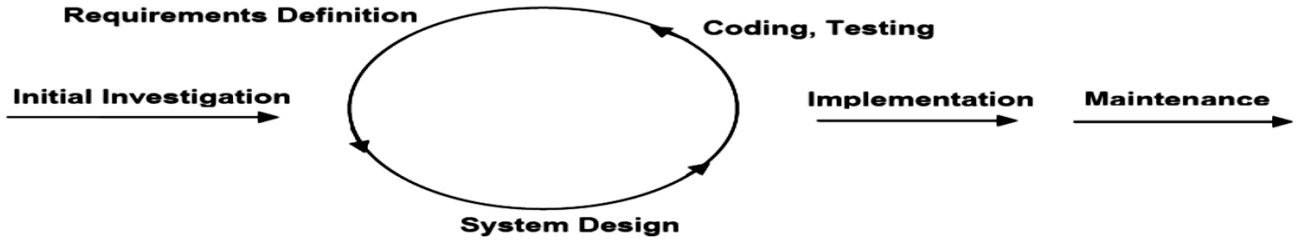
19. *THE PROTOTYPING MODEL [FRAMEWORK TYPE: ITERATIVE]*

PM

- The goal of **prototyping approach is to develop a small or pilot version of the new system called a prototype**. A prototype is a usable system or system component that is built quickly and at a lesser cost.
- Finally, when a prototype is developed that satisfies all user requirements, **either it is refined and turned into the final system or it is scrapped** and the knowledge gained from building the prototype is used to develop the real system.

Basic principles/generic phases

1) ***Step 1 - Identify Information System Requirements***:
2) ***Step 2 - Develop the Initial Prototype***:
3) ***Step 3 - Test and Revise***:
4) ***Step 4 - Obtain User Signoff of the Approved Prototype***:



Strengths:

1) Improves **user participation in system development**.
2) Especially useful for **resolving unclear objectives**.
3) Knowledge gained in an early iteration can be used in later iterations.
4) Helps to easily identify confusing/difficult and missing functionality.
5) Encourages innovation and flexible designs.
6) It typically results in a better definition of user needs and requirements in comparison to the traditional systems development approach.

PM

RTP

Weaknesses:

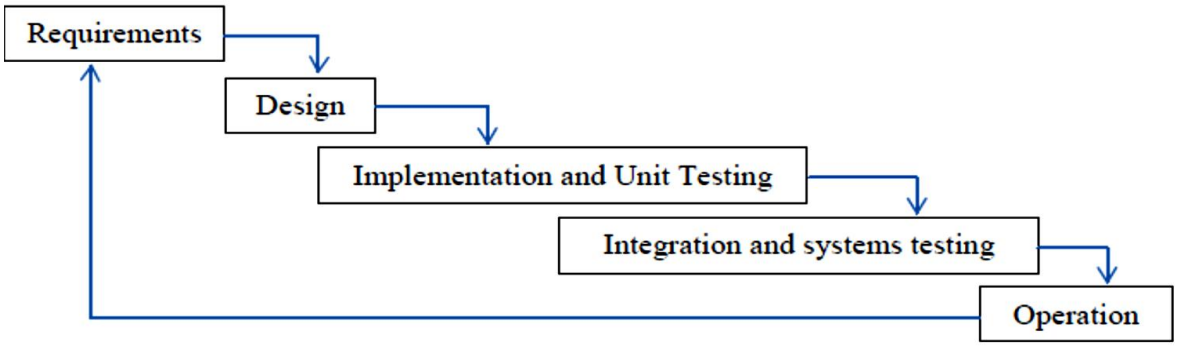
1) Approval process and control are not strict.
2) Incomplete or inadequate problem analysis may occur
3) Non-functional elements are difficult to identify and document.
4) Designers may prototype too quickly, without sufficient user needs analysis
5) Prototyping can be successful only if the system users are willing to devote significant time in experimenting with the prototype

© CA KUNAL AGRA

:: WWW.FACEBOOK.COM/PAGE.KUNAL :: KUNALSIR.IN ::
CONTACT # 9929503735/8233920041

20. THE INCREMENTAL MODEL

- Framework Type: Combination of Linear and Iterative.
- The product is decomposed into a number of components, each of which are designed and built separately (termed as builds). **Each component is delivered to the client when it is complete.**



Strengths:

1) Knowledge gained in an early increment can be used in the development of later increments.
2) Moderate control is maintained over the life of the project
3) Stakeholders can be given concrete evidence of project status.
4) More flexible.

Weaknesses:

1. Usually a lack of overall consideration of the business problem and technical requirements for the overall system.
2. Each phase of iteration is rigid and deals with specific issues related to that small part (build/increment) and do not takes into consideration other interdependent parts.
3. Problems may arise pertaining to overall system architecture because not all requirements are gathered up front for the entire software life cycle.

21. SPIRAL MODEL:

Pertinent Characteristics

1) The new system **requirements are defined in as much detail as possible**.
2) "Spiral Model" covers all possible alternatives, that can help in developing a cost effective project are analyzed and strategies are decided to use them.
3) A second prototype is evolved by a **fourfold procedure** :
    a. evaluating the first prototype
    b. defining the requirements
    c. planning and designing
    d. Constructing and testing

Strengths:

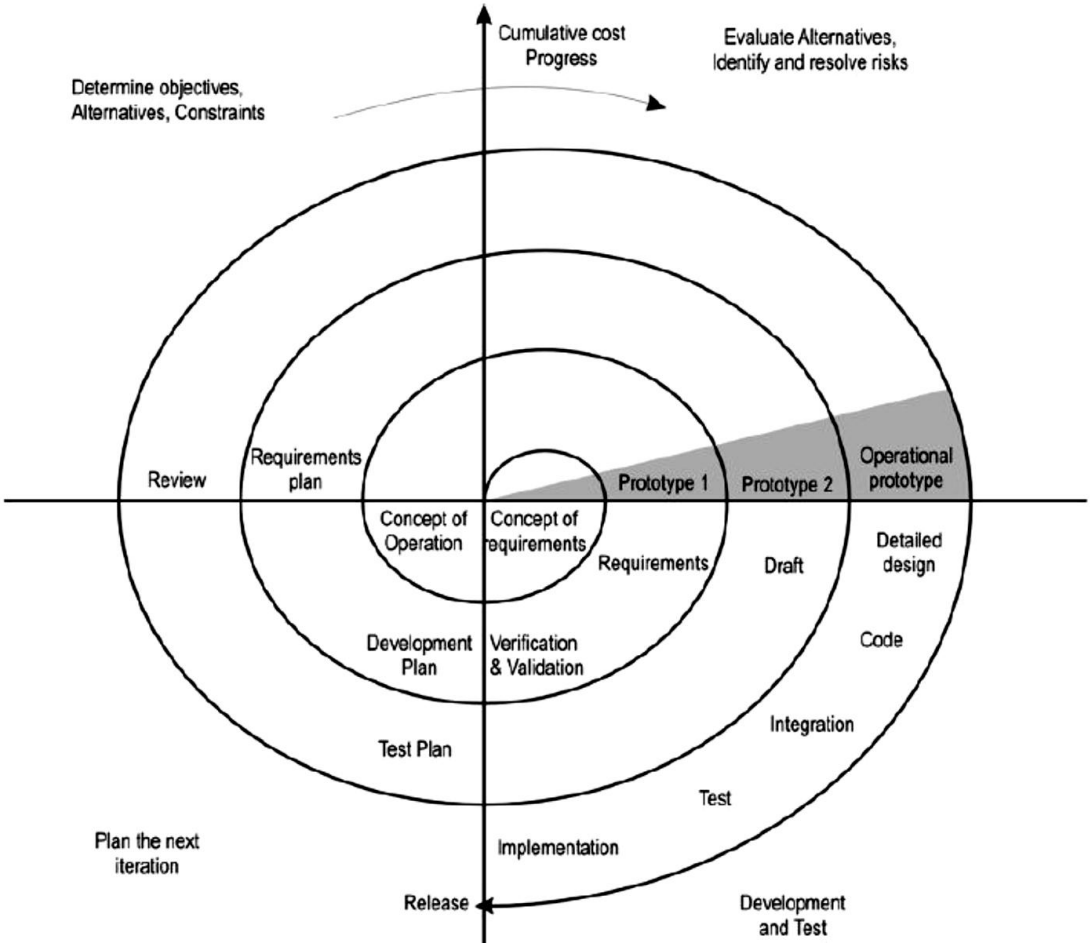
1) Low overall project risk.
2) Helps to select the best methodology
3) Can incorporate Waterfall, Prototyping, and Incremental methodologies. For example, a project with low risk of not meeting user requirements but high risk of missing budget or schedule targets would essentially follow a linear Waterfall approach for a given software iteration.

© CA KUNAL AGRAWAL

PM

Weaknesses:

1) **Challenges to determine the exact composition of development methodologies** to use for each iteration around the Spiral (i.e. use waterfall, prototype or incremental).
2) **Highly customized** to each project, and thus is **quite complex, limiting reusability**.
3) A **skilled and experienced project manager is required**
4) **No established controls for moving from one cycle to another cycle**.



22. *RAPID APPLICATION DEVELOPMENT (RAD)*

❖ Rapid Application Development (RAD) refers to a type of software development methodology which uses minimal planning. The "planning" work of software developed using RAD is included with writing the software itself.

- Key objective is for fast development and delivery of a high quality system at a relatively low investment cost,
- Aims to produce high quality systems quickly, primarily through the use of iterative Prototyping **Graphical User Interface(GUI) builders, Computer Aided Software Engineering (CASE) tools, Database Management Systems (DBMS),**
- Project control involves prioritizing development and defining delivery deadlines or "timeboxes."
- Produces documentation necessary to facilitate future development and maintenance.

Strengths

1) The operational version of an application is available much earlier
2) Because RAD produces systems more quickly and to a business focus
3) Quick initial reviews are possible.
4) Constant integration of small modules isolates problems and encourages user feedback.

Page 14 © CA KUNAL AGRAWAL

Weaknesses

1) More speed and lower cost may lead to a lower overall system quality.
2) Danger of misalignment of developed system due to missing information.
3) Project may end up with more requirements than needed (gold plating).
4) Tendency for difficult problems to be pushed to the future to demonstrate early success to management.

23. AGILE METHODOLOGIES

PM

Basic Principles:

1) Group of S/W development methodologies for fast development
2) Iterative and Incremental approach
3) Working S/W is delivered frequently (Build Short – Build Often)
4) Time Box
5) Close co-operation between user and developers
6) Face-to-face conversation
7) Rapid and flexible response to changes

Strength:

1) Adaptive, can respond to changing requirement
2) Face-to-face communication, no space for guess work
3) Minimum documentation saves time
4) High quality S/W in minimum time

PM

Weakness:

1. Difficult to assess the efforts required for S/W development at the beginning
2. No importance on necessary design and documentation
3. Verbal communication results in less knowledge transfer for future reference
4. More re-work required
5. Project deviates if user in not clear about final objective/outcome
6. Experienced team is required
7. Lacks attention to outside integration

PM

Other topic from PM

If you are the Project Manager of a Software Company with the responsibility for developing a break-through product, combining state of the art hardware and software; will you opt for prototyping as a process model for a product meant for the intensely competitive entertainment market?

Prototyping as a process model will be inappropriate and hence inadvisable for the following reasons:

- ✓ Prototyping requires user involvement. Here, users are consumers of the product who are diffused and may not be inclined to join in.
- ✓ When we try to test the product with the involvement of customers, confidential or critical information might get leaked to the competitors on our line of thinking. The element of surprise and also the opportunity to capture the market will be lost.
- ✓ Prototyping requires significant time for experimenting. Since the product is meant for the intensely competitive entertainment market, the project manager may not have that much time to experiment, and the competitor may capture the market by entering the market in advance.

© CA KUNAL AGRAWAL