TRINITY INTERNATIONAL COLLEGE

CASE STUDY 1:

OVERVIEW OF SYSTEM ANALYTICS AND DESIGN

Samrajya Chand
7/26/2023
CSIT 5th SEM

SYSTEM

A system is an orderly grouping of interdependent components working together to achieve a specific goal.
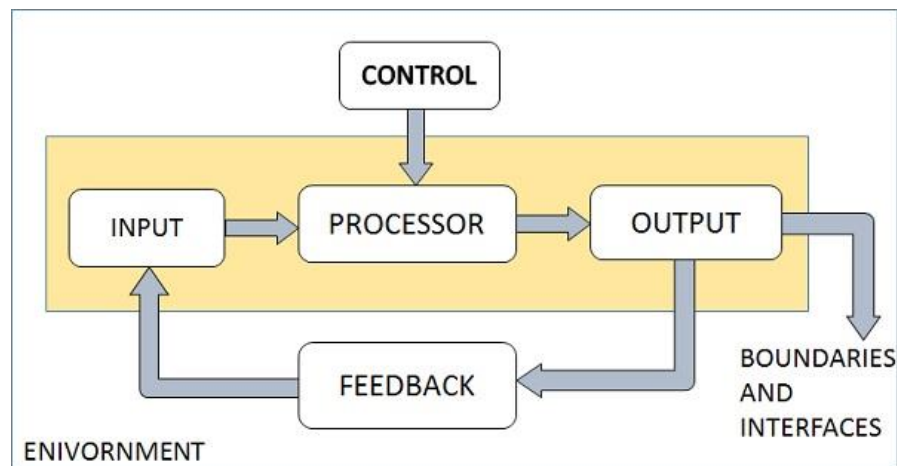

FIG: ELEMENTS OF SYSTEM

1.1. ELEMENTS OF SYSTEM

a) Inputs and Output
- Information entered in a system is input to the system.
- Output is the outcome of input after processing.
- The main aim of a system is to produce an output which is useful for its user.

b) Processor
- Transforms user input into useful output.
- Operational component of system.
- Processors may modify the input either totally or partially, depending on the output specification.

c) Control
- It Guides the System.
- It is the decision–making subsystem that controls the pattern of activities governing input, processing, and output.

d) Feedback

- Output is checked with desired output & necessary steps taken to achieve desired output.
- Positive feedback is routine in nature that encourages the performance of the system. Negative feedback is informational in nature that provides the controller with information for action.
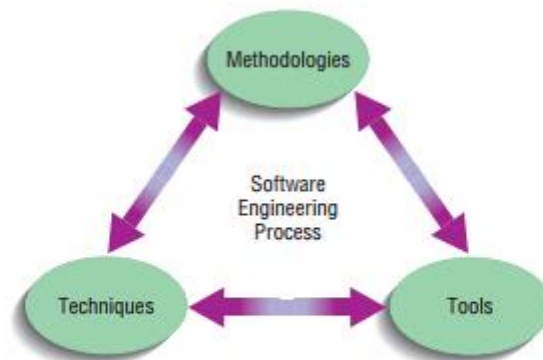
e) **Boundaries and Interface**
- Boundaries are the limits under which components of system interacts with each other.
- Each system has boundaries that determine its sphere of influence and control.
- Interconnection and interaction between subsystems is the interface.
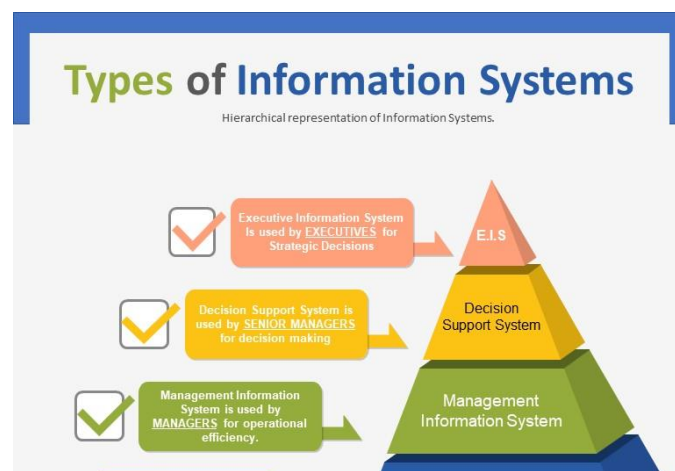
f) **Environment**
- Things outside boundary of system are Environment.
- It is source of external elements that strike on system.
- Change in environment affects the working of system.

2. INFORMATION SYSTEM

Information systems are a set of interconnected elements working together to collect, process, store, and distribute information to help coordination, visualization in an organization, analysis, and decision-making.

2.1. TYPES OF INFORMATION SYSTEM

Types of Information Systems

Hierarchical representation of Information Systems.

Executive Information System is used by EXECUTIVES for Strategic Decisions — E.I.S

Decision Support System is used by SENIOR MANAGERS for decision making — Decision Support System

Management Information System is used by MANAGERS for operational efficiency. — Management Information System

1. Transaction Processing System
   - Information system that processes data resulting from the occurrences of business transactions
   - Their objectives are to provide transaction in order to update records and generate reports i.e. to perform store keeping function.

2. Management Information System
   - Management Information System is designed to take relatively raw data available through a Transaction Processing System and convert them into a summarized and aggregated form for the manager, usually in a report format.

3. Decision Support System
   - Decision Support System is an interactive information system that provides information, models and data manipulation tools to help in making the decision in a semi-structured and unstructured situation.
   - Similar to MIS, but the end user is much involved in its actual creation.

4. Experts System
   - An expert system is an extension of DSS that captures and reproduces the knowledge and expertise of an expert problem solver or decision maker and then simulates the "thinking" or "actions" of that expert.

5. Office Automation System
   - Office automation (OA) is more than word processing and spreadsheet applications it support the wide range of business office activities for improved work flow and communication between workers, regardless of whether or not those workers are located in the same office.

6. Group Decision Support System
   - Consists of interactive software that allows for making decisions by a group of participants. The goal of a GDSS is to improve the productivity of a group to come to a decision.

7. Computer Supported Collaborative Work System
   - Computer-supported cooperative work (CSCW) describes coordinated activities performed by a group of participants in order to reach a common result supported by a computer system.

8. Executive Support System
   - Executive Support System (ESS) also called as Executive Information System (EIS) is a special purpose information system that supports executive decision-making. It is used primarily by top management.

3.1 System Analysis and Design

- Process of gathering facts, designing problems and using it to improve the system is System Analysis.
- Defines what system should do.
- Process of planning a new system to replace or edit the old one.
- Defines how to achieve the goal.
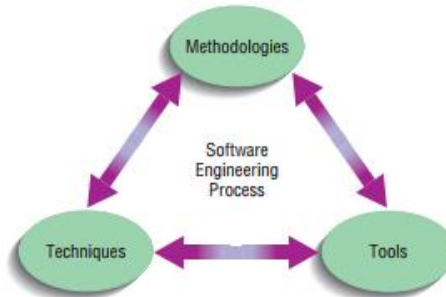
- Methodologies are a sequence of step-by-step approaches that help develop your final product:
- Tools are computer programs, such as computer-aided software engineering (CASE) tools, that make it easy to use specific techniques.
- Techniques are processes that you, as an analyst, will follow to help ensure that your work is well thought-out, complete, and comprehensible to others on your project team.

3.2. System Analyst

- System analyst is responsible for analyzing, designing and implementing systems to fulfill organizational needs. He/she plays a vital role in making operational the management information system

3.3. Roles of System Analyst

i) As Consultant

- May Act as a consultant to humans & businesses to address information system issues in a business.
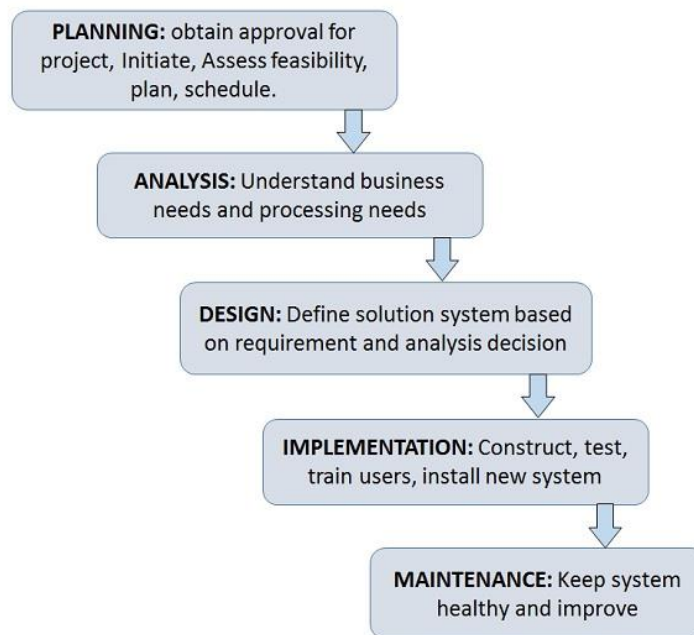- Gives fresh perspective that other people might not possess.

ii) As Supporting Agent

- May play supporting expert in some businesses we are hired for
- Draws on professional expertise concerning computer software & hardware uses.
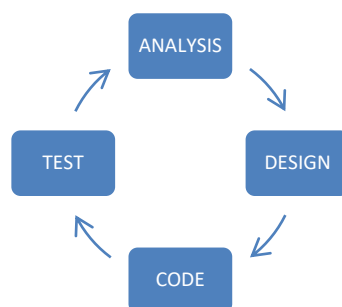
Iii) As an Agent of Change

- Systems analysts may serve as change agents who identify the organizational improvements needed, design systems to implement those changes and train and motivate others to use the systems.

4.1 DEVELOPMENT INFORMATION SYSTEM & SDLC

- Most organizations use a standard set of steps, called a systems development methodology to develop and support their information systems.
- It includes different phases as shown in the figure below. This representation of SDLC is sometimes referred to as the waterfall model or classic life cycle.

PLANNING: obtain approval for project, Initiate, Assess feasibility, plan, schedule.

ANALYSIS: Understand business needs and processing needs

DESIGN: Define solution system based on requirement and analysis decision

IMPLEMENTATION: Construct, test, train users, install new system

MAINTENANCE: Keep system healthy and improve

4.2 HEART OF SYSTEM DEVELOPMENT PROCESS

ANALYSIS

DESIGN

CODE

TEST

These activities are heart of the system development all the activities are combined into a single analysis-design-code-test process.

5.1 TRADITIONAL WATERFALL SDLC

- Also called classic lifecycle is the oldest and most widely used paradigm for information systems development.
- Formalized step by step approach consists of phase or activities and looks from top-down view
- Activities of one phase must be completed before moving to the next one as it can't be redone once started.
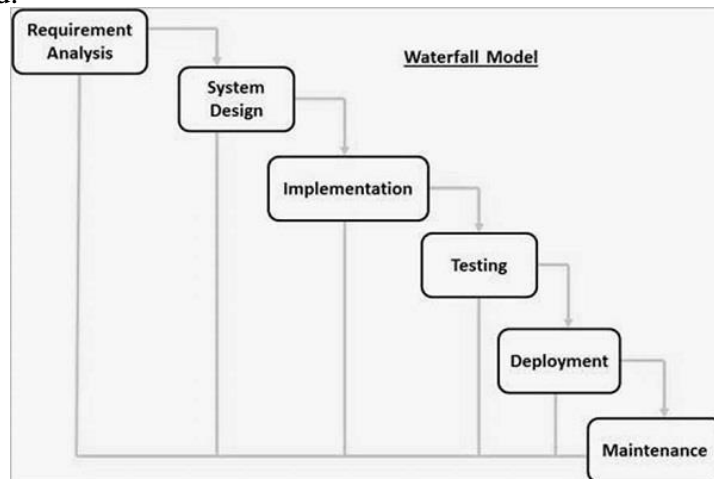
6.1 CASE-TOOLS

- CASE also called computer-aided software engineering is a technique using CASE tools to help system analyst to develop and maintain information systems.

Types of CASE Tools (Contd..) –Examples-

- Diagramming – for representing processes, data and control structures graphically. (analysis)
- CASE repository – holds information required to create, modify and evolve the system. (analysis, design, implementation)
- Form and report generators – automate generation of forms and reports to aid prototyping. (design, implementation, RAD, XP)
- Code generators – automate generation of source code from diagrams and forms. (design, implementation)
- Project management – aid in the planning, tracking, controlling and reporting of project management. (planning)
- Document generator – create standard reports based upon the contents of the CASE repository. (analysis, design, implementation)
- CASE analysis tools – help to identify problems of inconsistency, redundancy, and omissions. (more likely in analysis and design)

2 ways:

Forward engineering: Drawing system model and converting into program code.

Backward engineering: Reading existing code and converting into respective system model.

7.1 OTHER APPROACHES

- Different Approaches that has been developed to improve system analytics and design. They are:

7.2 PROTOTYPING APPROACH

- Useful for those whose project requirement is not fully known or constant update is required.
- Considered as Trial and Error Method
- Divided into 2 types
  a) Rapid throwaway Prototyping: Prototype doesn't need to be final one & can be iterated to develop new versions of final project.
  b) Evolutionary Prototyping: Developed prototype is the final model which is accepted after customers opinion satisfies.
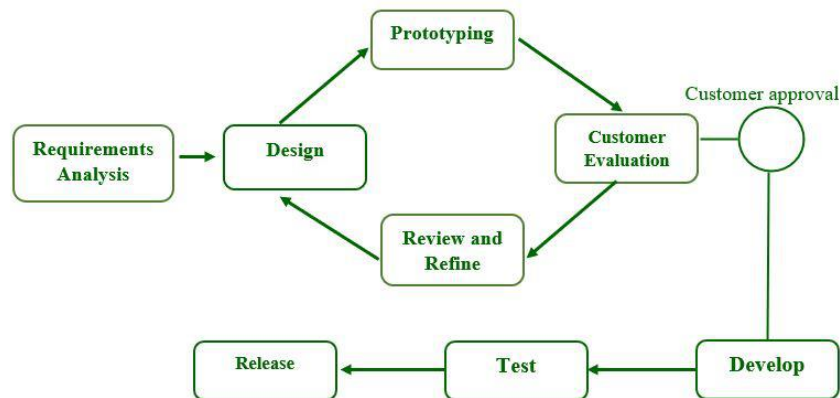
7.3 SPIRAL APPROACH

- The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model.
- It is the waterfall model with a very high emphasis on risk analysis.
- It looks like a spiral with many loops but exact number of loops can vary.
- Its radius determines the cost of the project & the angular dimension its progress so far.
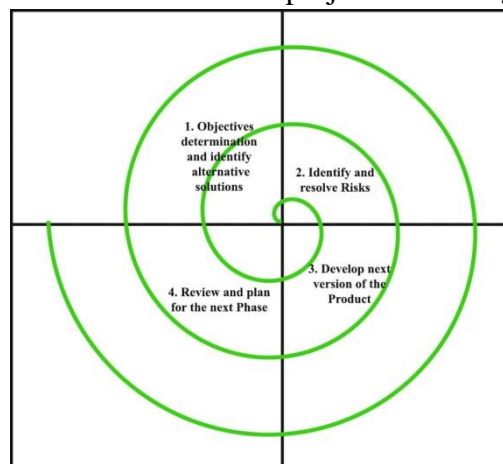
Advantages	Disadvantages
High amount of risk analysis so less risk	Costly model
Strong approval & documentation control	Risk analysis requires highly specific expertise.
Additional functional can be added	Doesn't work for small projects

7.4 RAPID APPLICATION DEVELOMENT APPROACH

- Similar to prototyping as they are conceptually close
- It is a software development methodology that uses minimal planning in favour of rapid prototyping
- The most important aspect for this model to be successful is to make sure that the prototypes developed are reusable.

Phases:

a) Requirements Planning Phase

- In this phase analyst & users meet to identify objectives of system & identify information requirements arising from those objects

b) RAD design workshop

- It is a design and refine phase.
- Users respond to actual working prototypes & analyst refine designing modules based on user responses.

c) Implementation phase

- Analyst & user work together to design business or non-technical aspects of system.
- As soon as these aspects are built it is tested & introduced to the organization.
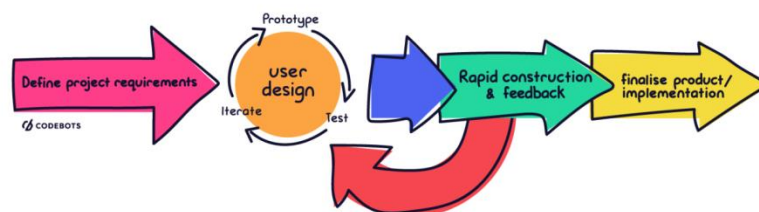
7.5 AGILE DEVELOPMENT PROJECT

- Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.

- The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.
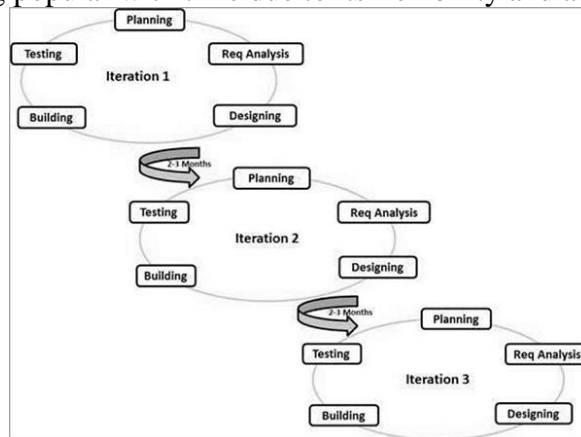
Following are the Agile Manifesto principles −

- Individuals and interactions − In Agile development, self-organization and motivation are important, as are interactions like co-location and pair programming.
- Working software − Demo working software is considered the best means of communication with the customers to understand their requirements, instead of just depending on documentation.
- Customer collaboration − As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements.
- Responding to change − Agile Development is focused on quick responses to change and continuous development.

7.6 EXTREME PROGRAMMING

- Extreme programming (XP) is one of the most important software development frameworks of agile models.
  It is used to improve software quality and responsiveness to customer requirements.
- Code Review: Code review detects and corrects errors efficiently. It suggests pair programming as coding and reviewing of written code carried out by a pair of programmers who switch their works between them every hour.
- Testing: Testing code helps to remove errors and improves its reliability. XP suggests test-driven development (TDD) to continually write and execute test cases. In the TDD approach, test cases are written even before any code is written.
- Incremental development: Incremental development is very good because customer feedback is gained and based on this development team comes up with new increments every few days after each iteration.
- Simplicity: Simplicity makes it easier to develop good quality code as well as to test and debug it.
- Design: Good quality design is important to develop good quality software. So, everybody should design daily.

- Integration testing: It helps to identify bugs at the interfaces of different functionalities. Extreme programming suggests that the developers should achieve continuous integration by building and performing integration testing several times a day.

7.7 SERVICE ORIENTED ARCHITECTURE

- SOA is an architectural approach in which applications make use of services available in the network. In this architecture, services are provided to form applications, through a network call over the internet.
- It defines a way to make software components reusable using the interfaces.
- In order to set up SOA,services must be:
  Modular, Reusable, Interoperability, Able to be categorized and identified, Able to be monitored, comply with industry specific standards

Formally, SOA is an architectural approach in which applications make use of services available in the network. In this architecture, services are provided to form applications, through a network call over the internet.
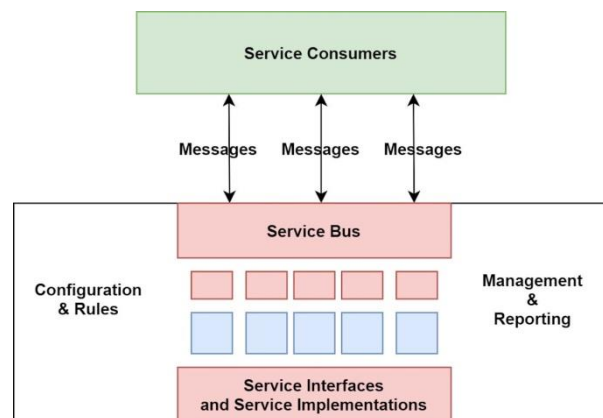

8 OBJECT ORIENTED ANALYSIS AND DESIGN

- Object-Oriented Analysis and Design (OOAD) is a software engineering methodology that involves using object-oriented concepts to design and implement software systems.
- OOAD involves a number of techniques and practices, including object-oriented programming, design patterns, UML diagrams, and use cases.

9. MANAGING THE INFORMATION SYSTEM PROJECT

- In this section, we focus on system analyst's role as project manager of an information system project.
- Project management is an important aspect of development of information system and a critical skill for system analyst
- A project is a planned undertaking of a series of related activities to reach an objective that has a beginning and end
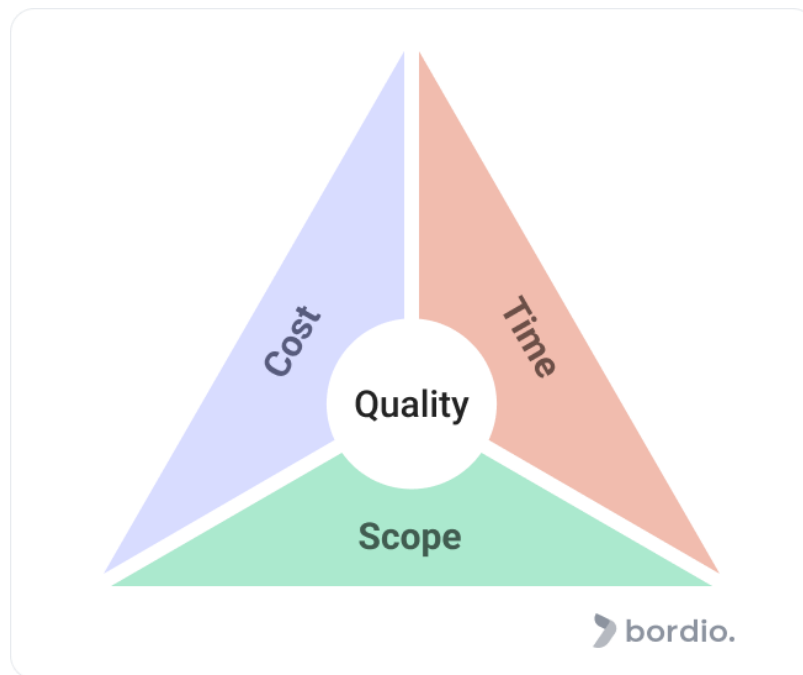
- A project manager is a professional who organizes, plans, and executes projects while working within restraints like budgets and schedules.

The top 5 skills of a successful
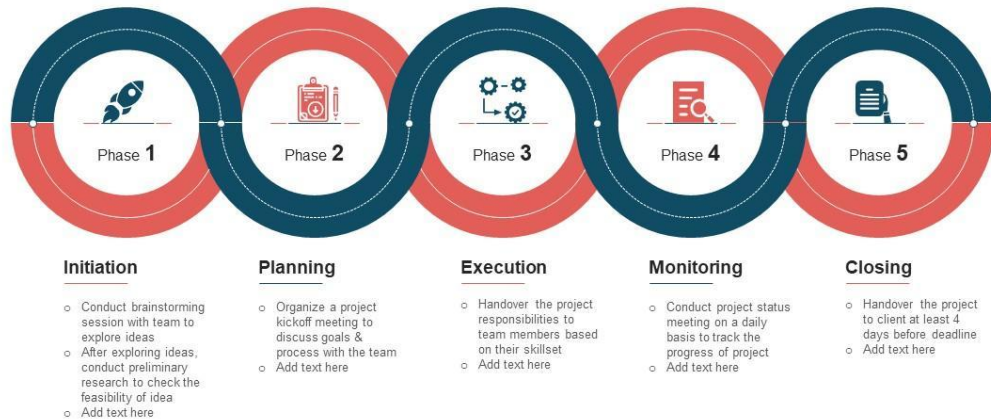**PROJECT MANAGER**

The initial challenge with managing any project is to achieve all of the project objectives within the given restrictions. Here are our top 5 fundamental project management skills required to run projects efficiently.

1 Communication	2 Organization	3 Negotiation	4 Mentorship Mentality	5 Technological Skills
Excellent communication skills are essential for anyone undergoing a project management role. Statistics show that more than half of all projects fail due to ineffective communication.	A project manager can only succeed if their organizational skills are up to scratch. They have to co-ordinate different reports, deadlines, tasks, meetings, risks and QA issues all in a day's work.	The skills involved with negotiation are required throughout a project, not only at the beginning. A manager needs to know when to apply formal and informal negotiations to the appropriate situations.	A good project manager has the ability to motivate their team and act as a mentor. They need to lead by example and maintain the correct balance between serving their team and serving their company.	The best project manager will have the ability to complete tasks themselves, as well as delegating. This earns them respect from their colleagues and an understanding of what they are asking of their team.

10. PHASES OF PROJECT MANAGEMENT

Five Phases of Project Management Life Cycle

Following slide outlines 5 phases of project management life cycle that a project manager can use. Initiation, planning, execution, monitoring and closing are the core phases which will assist managers in completing projects successfully.

Phase 1 **Phase 2** **Phase 3** **Phase 4** **Phase 5**

Initiation
- Conduct brainstorming session with team to explore ideas
- After exploring ideas, conduct preliminary research to check the feasibility of idea
- Add text here

Planning
- Organize a project kickoff meeting to discuss goals & process with the team
- Add text here

Execution
- Handover the project responsibilities to team members based on their skillset
- Add text here

Monitoring
- Conduct project status meeting on a daily basis to track the progress of project
- Add text here

Closing
- Handover the project to client at least 4 days before deadline
- Add text here

This slide is 100% editable. Adapt it to your needs and capture your audience's attention.

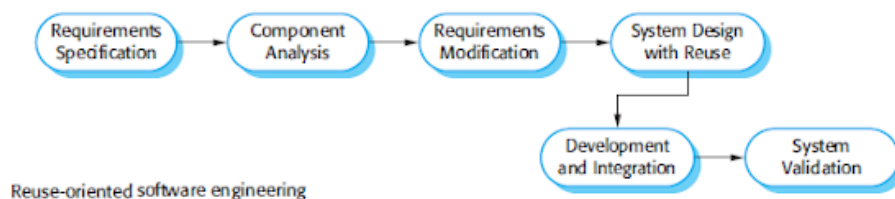## 11. CALCULATING EXPECTED TIME USING PERT

ET= (0+4r+p)/6

Where, ET= expected time for completion for an activity

O=optimistic completion time

r = realistic completion time

p=pessimistic completion time

12. REUSE

Reuse-oriented software engineering

- It is the use of previously written software resources in new application
- The reuse-oriented model isn't always sensible in its pure form due to cause of an entire repertoire of reusable additives that might not be available.
- Reuse-oriented approaches rely on a large base of reusable software components and an integrating framework for the composition of these components.