Introduction to system Analysis and Modeling

CHAPTER ONE

Chapter 1
Introduction to System Analysis and Modeling

- Defining a System
- Characteristics of a system
- Elements of a system
- Constraints of a System
- Types of systems
- System development methodologies (RAD, Agile)
- System models

System Analysis vs System Design

System analysis

- is a process for **reviewing a technological system for troubleshooting, development, or improvement**. Such a system might be a system or application program.
  - Helps to study a system or its parts in order to identify its objectives.
  - It specifies **what** the system should do.
  - It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components.

…Cont'd

System design

- The process of **defining the architecture, interfaces, and data model** for a system to satisfy the requirements outlined in the SRSs.
- At this stage, software engineers **translate business requirements into technical specifications** to build a new physical system or update an existing one

What is a System?

- **System**

  - **is “an orderly grouping of interdependent components linked together according to a plan to achieve a specific goal.”**
  - **A system is a collection of elements related in a way that allows a common objective to be accomplished.**

Constraints of a System

- **A system must have three basic constraints**
  - A system must have some **structure and behavior** which is designed to achieve a predefined objective.
  - **Interconnectivity** and **interdependence** must exist among the system components.
  - The **objectives of the organization** have a **higher priority** than the objectives of its subsystems.
    - **Example**, traffic management system, payroll system, automatic library system, human resources information system.

Characteristics of a System

Organization

- Organization implies structure and order.

- It is the **arrangement of components** that helps to achieve predetermined objectives.

Interaction

- It is defined by the manner in which the components operate with each other.
- **E.g.,** in an organization, purchasing department must interact with production department and payroll with personnel department.

…Characteristics of a System

- **Interdependence**
  - Determines how the components of a system depend on one another.
  - For proper functioning, the components are coordinated and linked together according to a specified plan.
  - The output of one subsystem is the required by other subsystem as input.

…Characteristics of a System

Integration

- Integration is concerned with how a system components are connected together.
- It means that the parts of the system work together within the system even if each part performs a unique function.

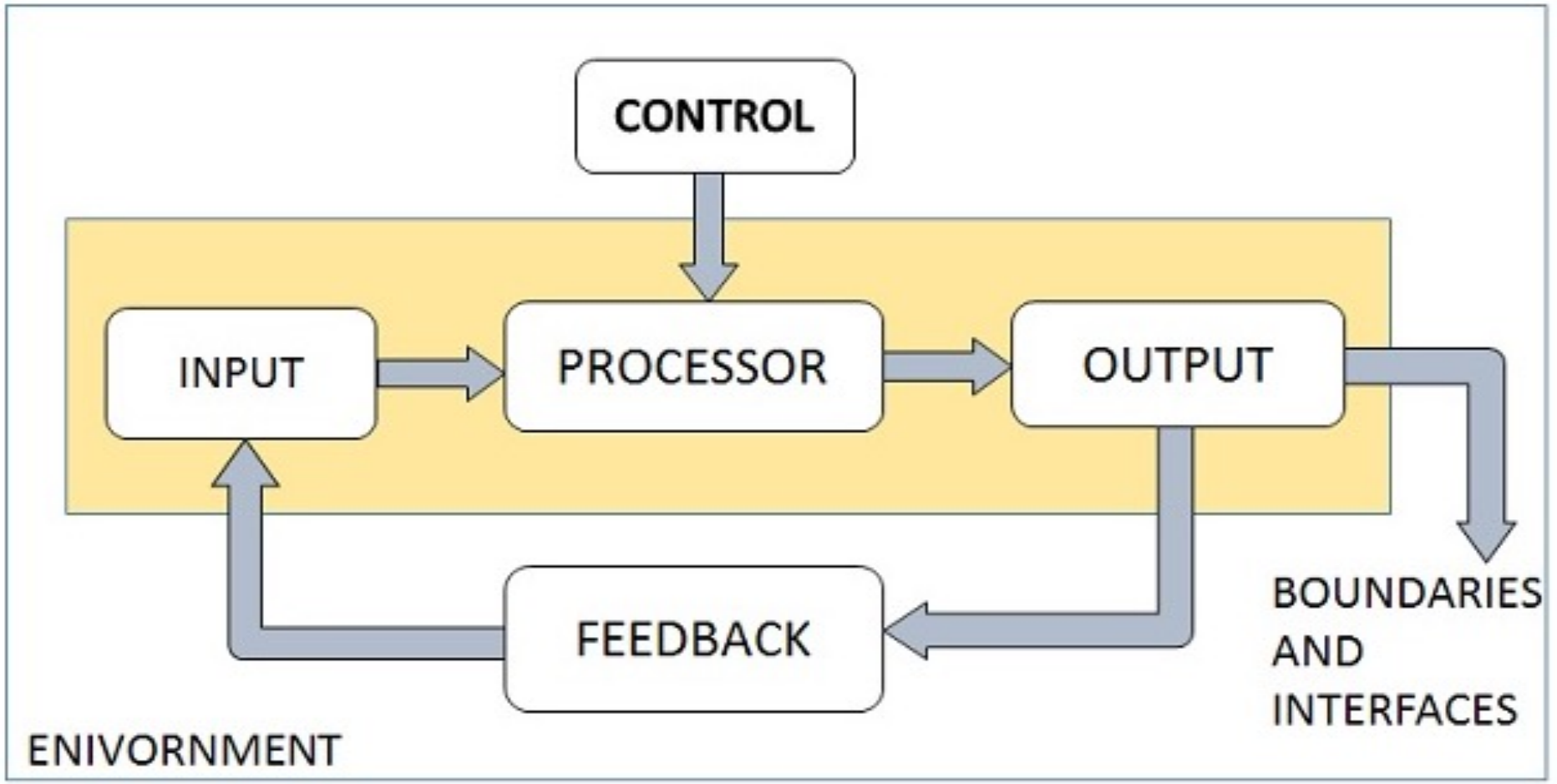
…Characteristics of a System

Central Objective

- The objective of system must be central. It may be real or stated.
- It is not uncommon for an organization to state an objective and operate to achieve another.
- The users must know the main objective of a computer application early in the analysis for a successful design and conversion.

Elements of a System

- Manger elements of every system :-



…Elements of a System

Outputs and Inputs

- The main aim of a system is to produce an output which is useful for its user.
- Inputs are the information that enters into the system for processing.
- Output is the outcome of processing.

…Elements of a System

Processor(s)

- The processor is the element of a system that involves the actual transformation of input into output. It is the operational component of a system.

- Processors may modify the input either totally or partially, depending on the output specification.

- As the output specifications change, so does the processing.

…Elements of a System

Control

- It is the decision–making subsystem that controls the pattern of activities governing input, processing, and output.
- **E.g**. the behavior of a computer System is controlled by the Operating System and software.
- In order to keep system in balance, what and how much input is needed is determined by Output Specifications.

…Elements of a System

Feedback

- Feedback provides the control in a dynamic system.
- 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.

…Elements of a System

Environment

- The environment is the "supersystem" within which an organization operates.
- The things outside the boundary of the system are known as environment.
- Change in the environment affects the working of the system
- It determines how a system must function.
- For example, vendors and competitors of organization's environment, may provide constraints that affect the actual performance of the business.

…Elements of a System

Boundaries and Interface

- **Boundaries** are the limits that identify its components, processes, and interrelationship when it interfaces with another system.
- Setting up boundaries helps for better concentration of the actives carried in the system.
- The interconnections and the interactions between the sub-systems is known as the **Interfaces.** They may be inputs and outputs of the systems.

Types of Systems

- Systems have been classified in different ways.
  - Common classifications are:
    - Physical or abstract systems
    - Open or closed systems
    - Deterministic or probabilistic systems
    - Man-made information systems

….Types of Systems

Physical or Abstract Systems

- Physical systems are tangible entities. We can touch and feel them.
- Physical System may be static or dynamic in nature.
  - **E.g.** desks and chairs are the physical parts of computer center which are static that facilitate operation of the computer.
- A program in a computer is a dynamic system in which programs, data, and applications can change according to the user's needs.

- Abstract systems are non-physical entities or **conceptual** that may be formulas, representation or model of a real system.

…..Types of Systems

Open or Closed Systems

- An open system must interact with its environment.
  - It receives inputs from and delivers outputs to the outside of the system.

    E.g. an information system which must adapt to the changing environmental conditions.
- A closed system does not interact with its environment.
  - It is isolated from environmental influences.
  - A completely closed system is **rare in reality.**

…….Types of Systems

❑Adaptive and Non Adaptive System

- **Adaptive System** responds to the change in the environment in a way to improve their performance and to survive.
  - **E.g.** human beings, animals.
- **Non Adaptive System** is the system which does not respond to the environment.
  - **E.g.,** machines.

❑Permanent or Temporary System

- Permanent System persists for long time. For example, business policies.
- Temporary System is made for specified time and after that they are demolished.
  - For example, A **DJ system is** set up for a program and it is dissembled after the program.

…..Types of Systems

Deterministic or Probabilistic System

- A **deterministic system** is one in which the occurrence of all events is known with certainty. If the description of the system state at a particular point of time of its operation is given, the next state can be perfectly predicted.
- Deterministic system operates in a predictable manner and the interaction between system components is known with certainty.
  - **E.g.** two molecules of hydrogen and one molecule of oxygen makes water.
- **Probabilistic System shows uncertain behavior.**
  - The exact output is not known.
  - **E.g**. Weather forecasting, mail delivery.
- A **probabilistic system** is one in which the occurrence of events cannot be perfectly predicted. Though the behavior of such a system can be described in terms of probability, a certain degree of error is always attached to the prediction of the behavior of the system.

…….Types of Systems

❑Social, Human-Machine, Machine System

- **Social System** is made up of people.
  - **E.g.**, social clubs, societies. nuclear family units, communities, cities, nations, college campuses, corporations, and industries.
- In **Human-Machine System**, both human and machines are involved to perform a particular task.
  - **E.g.,** Computer programming. HMIs utilized in the industrial context are mostly screens or touchscreens that connect users to machines, systems or devices.
- **Machine System** is where human interference is neglected. All the tasks are performed by the machine.
  - **E.g.,** an autonomous robot.

……Types of Systems

Man–Made Information Systems

- It is an interconnected set of information resources to manage data for particular organization, under Direct Management Control (DMC).
- This system includes hardware, software, communication, data, and application for producing information according to the need of an organization.

……Types of Systems

Information system (IS)

- A collection of Interrelated components that **collect, process** and **stored,** and provides as **output** the information needed to complete a business task.

Examples of IS

- Course Registration system
- Online order system
- Online Banking System, etc.

Types of Systems

- **Man-made information systems** are divided into three types −

a) Formal Information System − It is based on a very clear flow of information in the form of memos, instructions, etc., from top level to lower levels of management.

- represented by the **organization chart**. The chart is a map of positions and their authority relationships, indicated by boxes and connected by straight lines.
- It is concerned with the pattern of authority, communication and work flow. **E.g.** Strategic, managerial and operational information systems

Types of Systems

b) Informal Information System – This is employee based system which solves the day to day work related problems.

- **E.g.,** Conversation over coffee or launch, Message on social media or blogs, etc.

c) Computer Based System – This system is directly dependent on the computer for managing business applications.

- **E.g.,** automatic library system, highway systems, railway reservation system, airline systems, banking system, etc.

Systems Models

- System modeling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system.
- It is about representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML).

Systems Models

Schematic Models

- A schematic model is a 2-D chart that shows system elements and their linkages.
- Different arrows are used to show information flow, material flow, and information feedback.

Flow System Models

- A flow system model shows the orderly flow of the material, energy, and information that hold the system together.
- Program Evaluation and Review Technique (PERT), for example, is used to abstract a real world system in model form.

Systems Models

❑Static System Models

- ✓The static structural view of a problem, which does not vary with time.
- ✓It describes the static structure of the system being modeled, which is considered less likely to change than the functions of the system.
- ✓It includes class diagram and object diagrams and help in depicting static constituents of the system
  - ✓The Gantt chart, for example, gives a static picture of an activity-time relationship.

Example: Model of building.

Systems Models

Dynamic System Models

- Business organizations are dynamic systems.
- A dynamic model approximates the type of organization or application that analysts deal with.
- It shows an ongoing, constantly changing status of the system.
- Dynamic modelling on the other hand consists of sequence of operations, state changes, activities, interactions and memory.
- E.g. Queue Model in a bank System

System Development Methodologies

System Development methodologies

- **SDLC (Software Development Life Cycle)** is a process that consists of a series of planned activities to develop or alter the Software Products
  - It is a process used by the software industry to design, develop and test high quality software.
  - It is aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.
- **ISO/IEC/IEEE 12207** Systems and software engineering – Software life cycle processes is an international standard for software lifecycle processes.

… System Development methodologies

- There are various software development life cycle **models**. The most important and popular SDLC models followed in the industry

  i. **Waterfall Model**

  ii. **Iterative Model**

  iii. **Spiral Model**

  iv. **V-Model**

  v. **Big Bang Model (Reading Assignment) and Prototype model**

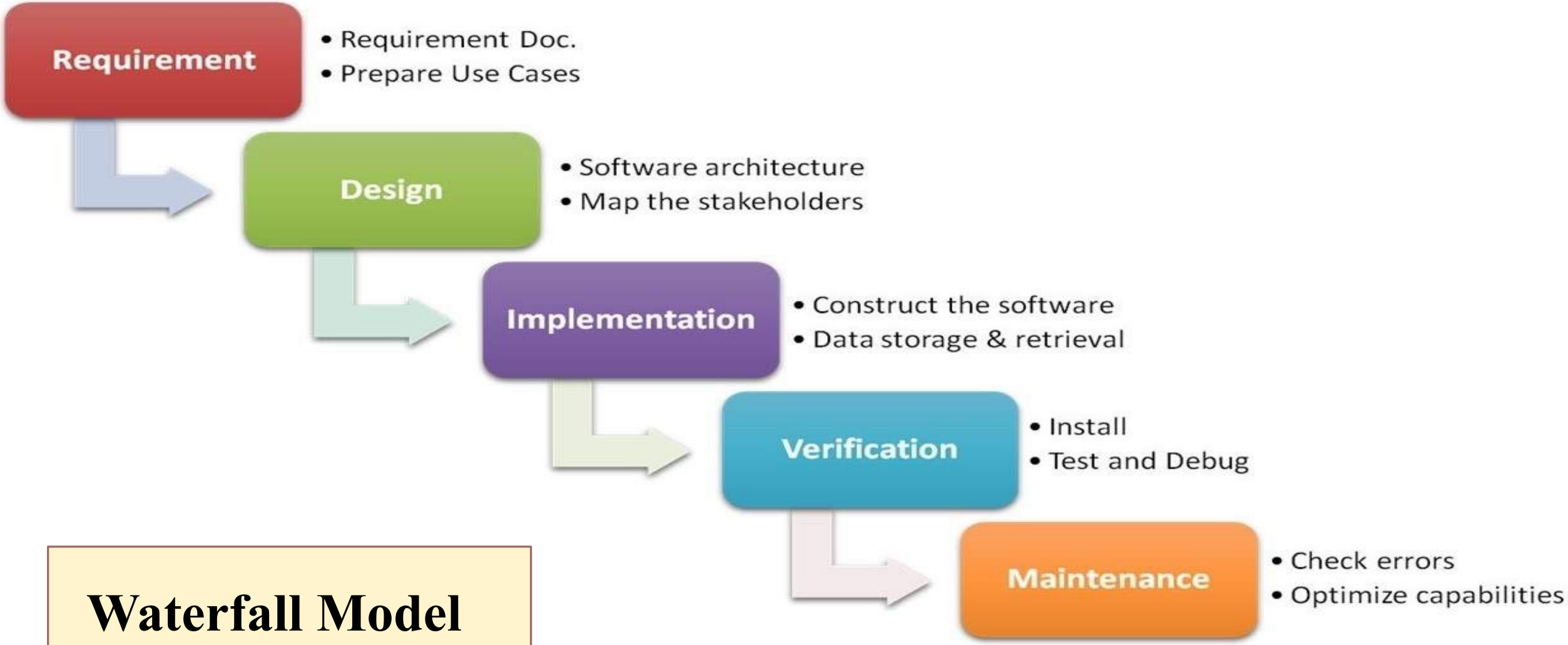
- Other related methodologies are **Agile Model, RAD Model**

… System Development methodologies

i. Waterfall Model

- It was the first SDLC Model to be used widely in Software Engineering to ensure success of the project.
- In "**The Waterfall**" approach, the whole process of software development is divided into separate phases in which the outcome of one phase acts as the input for the next phase sequentially.

… System Development methodologies



Waterfall Model

When to use SDLC Waterfall Model?

- Some Circumstances where the use of the Waterfall model is most suited are:
  - When the requirements are constant and not changed regularly.
  - A project is short
  - The situation is calm
  - Where the tools and technology used is consistent and is not changing
  - When resources are well prepared and are available to use.

… System Development methodologies

Advantages of Waterfall model

- This model is simple to implement also the number of resources that are required for it is minimal.
- The requirements are simple and explicitly declared; they remain unchanged during the entire project development.
- The start and end points for each phase is fixed, which makes it easy to cover progress.
- The release date for the complete product, as well as its final cost, can be determined before development.
- It gives easy to control and clarity for the customer due to a strict reporting system

… System Development methodologies

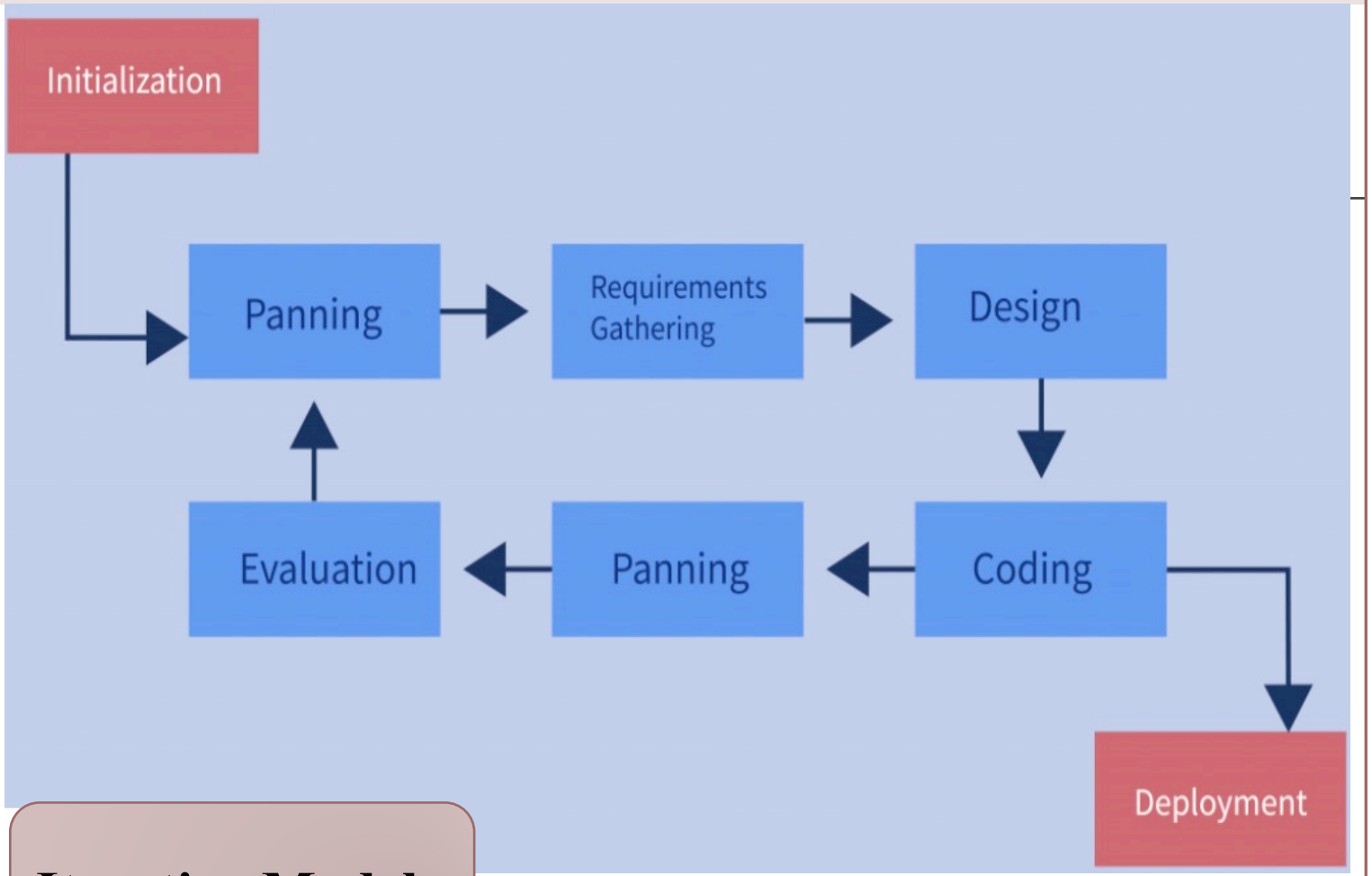
Disadvantages of Waterfall model

- The risk factor is higher, so this model is not suitable for more significant and complex projects.
- This model cannot accept the changes in requirements during development.
- It becomes tough to go back to the phase. For example, if the application has now shifted to the coding phase, and there is a change in requirement, It becomes tough to go back and change it.
- Since the **testing done at a later stage**, it does not allow identifying the challenges and risks in the earlier phase, so the risk reduction strategy is difficult to prepare.

ii. Iterative Model

- The initial development work is conducted **based on initial requirements that are clearly defined**, and subsequent features are added to this base software product through iterations until the final system is completed.
- This approach does not aim to create a broad specification plan. Instead, the iterative development model is a method for breaking down any major software development project into **smaller chunks.**
- Post that, the prototype is examined again for any extra requirements and then the rest of the planning, requirement analysis, deployment, and maintenance are all conducted. This helps in identifying risks associated with the requirements at a early stage and mitigate them.

… System Development methodologies



Iterative Model

Note:

- Allows us to spot any major design or planning issues in the process model and fix them as early as possible as this model is cyclic in nature.
- It is quite useful since it can accept modifications in the system's original requirements.
- can also be used in conjunction with other models such as the **incremental model, Agile methodology, and so on**.

… System Development methodologies

Some Features of the Iterative Model

- It enables you to demonstrate and measure the progress of your project without any bias.
- The functionality of the project increases Incrementally.
- It lets you have a constant improvement in the quality of the project.
- With continuous improvements, it helps you lower down the chances of risk.
- You can experience good improvement in the accuracy of the various estimates that could be part of the project.
- It is also known as the **cyclic model**. *After the initial phase, some phases occurs repeatedly and with the completion of each phase there is the scope of some improvement.*
- It helps increasing collaboration, and effectiveness within the team.

… System Development methodologies

Advantages Iterative Model

- In an iterative paradigm, less effort is spent documenting and more time is allocated to design.
- It is easily adjustable and flexible to the project's and client's changing requirements.
- In comparison to other process models, this paradigm is significantly less expensive to change requirements as we work on developing the project iteratively once the requirements are frozen.
- The end-user can swiftly provide input after each iteration, which can subsequently be incorporated into the system thereby improving the experience of the application.

… System Development methodologies

Disadvantages Iterative Model

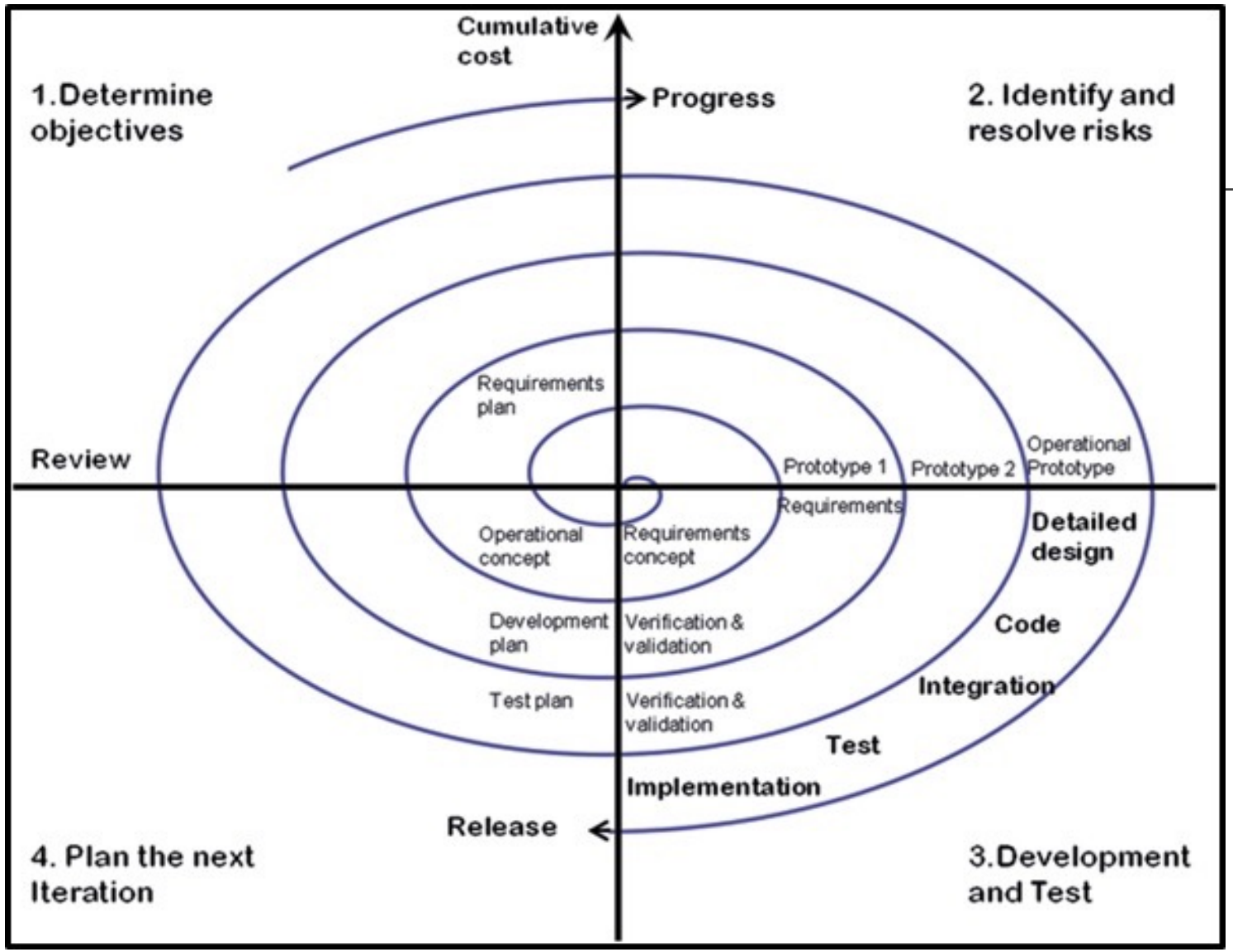
- There is a need for proper management.
- The iterative model is not the correct choice for small project as it may not be possible or realistic to break down small projects into more smaller parts.
- This model requires highly skilled resources to work on the analysis part of the project to avoid risk.
- Since all the requirements are not gathered well in advance, problems with the system design may arise.
- The entire procedure is difficult to manage.

… System Development methodologies

iii. Spiral model

- **Risk-driven** software development model. Means, it is most suitable for long term, complex and high risk project.
- In its diagrammatic representation, it looks like a spiral with many loops.
- The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a **Phase of the software development process.**
- As the project manager dynamically determines the number of phases, so the project manager has an important role to develop a product using the spiral model.

… System Development methodologies



The Spiral Model

… System Development methodologies

Each phase of the **Spiral Model** is divided into four quadrants and the functions of these four quadrants are discussed below-

---

1. Objectives determination and identify alternative solutions: Requirements are gathered from the customers and the objectives are identified, elaborated, and analyzed at the start of every phase. Then alternative solutions possible for the phase are proposed in this quadrant.

2. Identify and resolve Risks: During the second quadrant, all the possible solutions are evaluated to select the best possible solution. Then the risks associated with that solution are identified and the risks are resolved using the best possible strategy. At the end of this quadrant, **the Prototype is built for the best possible solution.**

… System Development methodologies

3.Develop next version of the Product: During the third quadrant, the identified features are developed and verified through testing. At the end of the third quadrant, the next version of the software is available.

4. Review and plan for the next Phase: In the fourth quadrant, the Customers evaluate the so far developed version of the software. In the end, planning for the next phase is started.

… System Development methodologies

Risk Handling in Spiral Model

A risk is any adverse situation that might affect the successful completion of a software project.

- The most important feature of the spiral model is handling these unknown risks after the project has started.
- The spiral model supports coping up with risks by providing the scope to build a prototype at every phase of the software development.
- The Spiral model is called a **Meta-Model** because it incorporates all the other SDLC models **Iterative Waterfall Model**, **Prototyping Model**, **Evolutionary model**

… System Development methodologies

Advantages of Spiral Model

Below are some advantages of the Spiral Model.

- **Risk Handling:** The projects with many unknown risks that occur as the development proceeds, in that case, Spiral Model is the best development model to follow due to the risk analysis and risk handling at every phase.
- **Good for large projects:** It is recommended to use the Spiral Model in large and complex projects.
- **Flexibility in Requirements:** Change requests in the Requirements at later phase can be incorporated accurately by using this model.
- **Customer Satisfaction:** Customer can see the development of the product at the early phase of the software development and thus, they habituated with the system by using it before completion of the total product.

… System Development methodologies

Disadvantages of Spiral Model

Below are some main disadvantages of the spiral model.

- **Complex:** The Spiral Model is much more complex than other SDLC models.
- **Expensive:** Spiral Model is not suitable for small projects as it is expensive.
- **Too much dependability on Risk Analysis:** *The successful completion of the project is very much dependent on Risk Analysis.* Without very highly experienced experts, it is going to be a failure to develop a project using this model.
- **Difficulty in time management:** As the number of phases is unknown at the start of the project, so time estimation is very difficult.

… System Development methodologies

iv. V Model

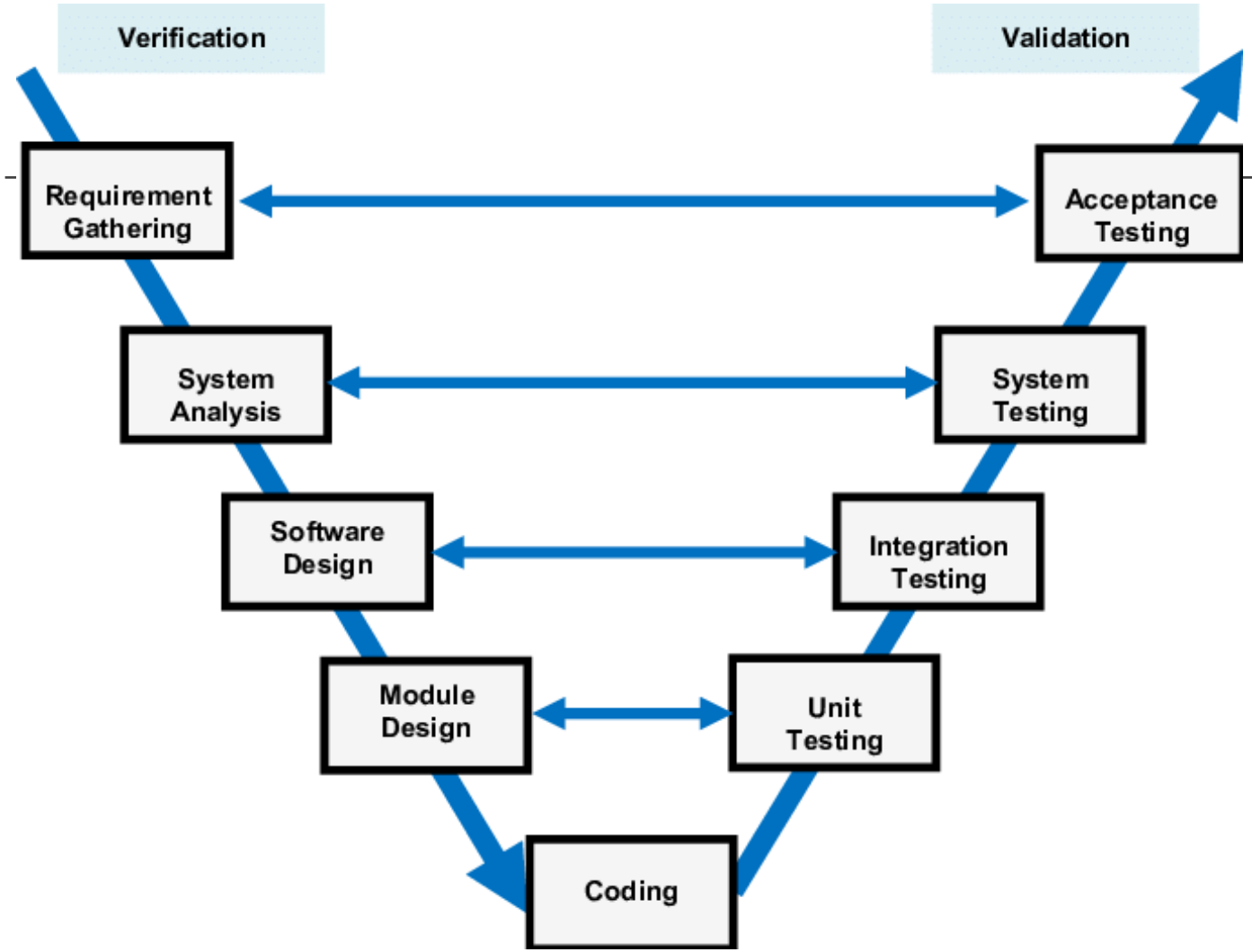
- V- model is also called **Verification** and **Validation** model .
- This model is the extension of the Waterfall Model.
- The V form of the V-model shows the various phases of the verification and validation phases.
- In this model one phase for verification and other for validation and the coding phase joins the both phases verification and Validation .So that makes the V shape so this model is called **V-model.**

… System Development methodologies

iv. V-Model

- A Software development Life Cycle (SDLC) that emphasizes the concept of "**Verification and Validation**" .
- In each step of development in **V-Model,** there will be a corresponding testing phase that will be validating such a process.
- The next phase of development start after completing first phase.
- Testing Phases will be planned in parallel with the development of the stage which they are supposed to be tested against and will be joined at the bottom by the actual coding process. **Hence the name V-Model.**

… System Development methodologies



V-Model

… System Development methodologies

❖V model model consists two main phases :

1. **Verification :**

➢The process to verify that **the software product development phase** to determine that specified requirements meet or not ? In this phase, there is no need to execute the code for testing.

2. Validation :

➢Validation is the process to verify that the software product fulfills the customer requirements and expectations or not. **In this phase, there is need of execution of the code.**

… System Development methodologies

Advantages of V Model :-

1- It works very well for small project according to their requirement .

2- This model is very simple, easy and useful.

3- This is a high quality model and all the phases are completed at once.

4- This model is use to track the process of project management .

5- This model saves a lot of time and efforts.

6- Testing is starting at the initial phase so there is no issue of bugs.

7- Client’s requirements are not clearly specified.

… System Development methodologies

Disadvantages of V-Model :-

1- This model can not be use for large project.

2- This model is not good if customer's requirements are not clear.

3- There are lots of risk.

4- This model is not easy for complex projects .

5- Client have no prototype and involvement during the software development.

6- This model contains less flexibility.

7- It is hard to go back and alter the working of the system if new requirements are met.

… System Development methodologies

Agile Model

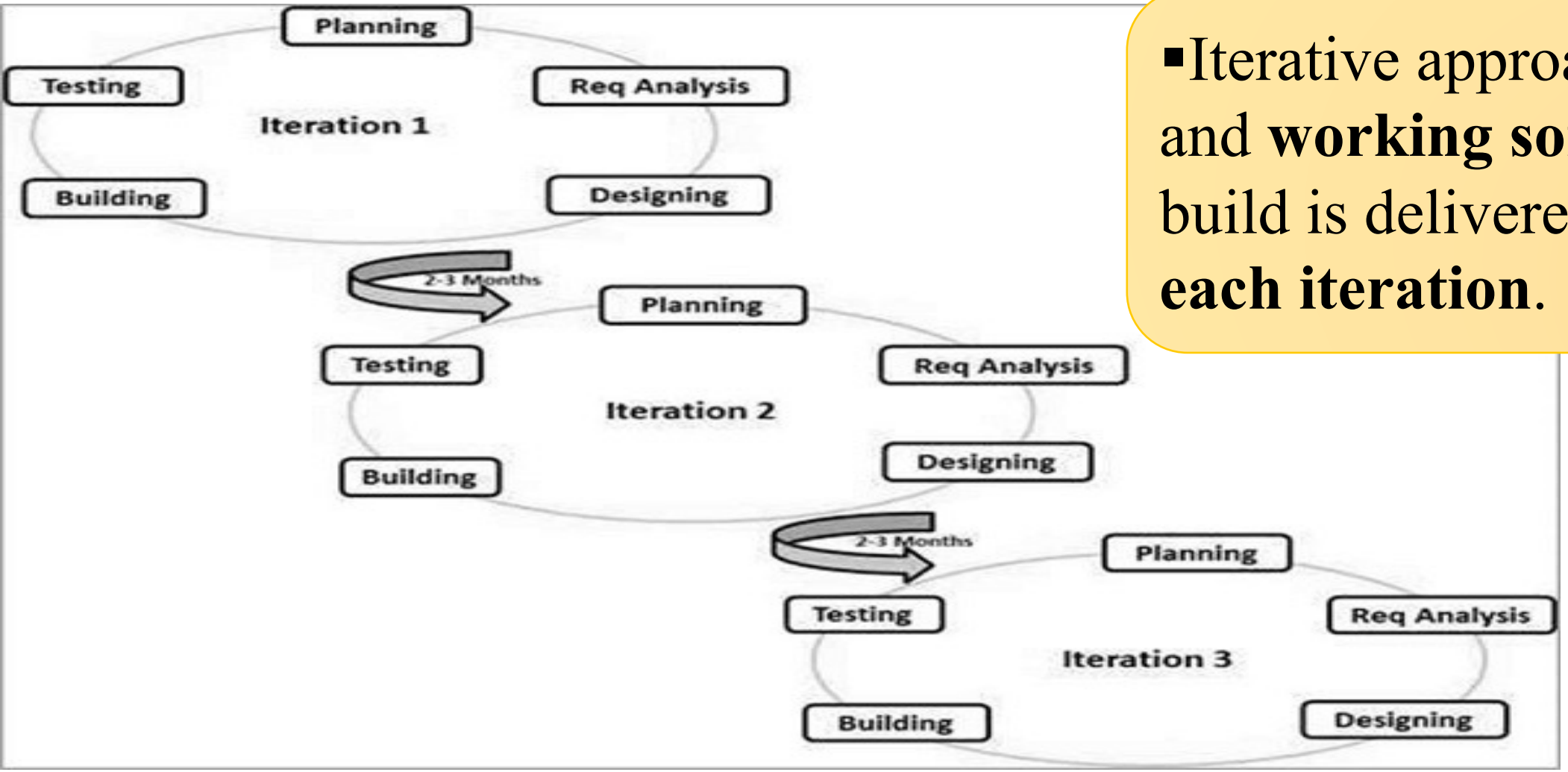
- Overcome the limitation of the waterfall model,
- It is the combination of **iterative and incremental** software development model.
- In this model the requirements are break up into many parts, called **iterations**, and then developed incrementally.
- Each iteration is planned, designed, implemented, tested and deployed to the customers to take the feedback.
- If any changes required then the modification is done at that iteration then carry on the project.
- Any error can be fixed at each iteration so there is no issue about presence of errors in the project.

… System Development methodologies

- The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.
- Each iteration(module) involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.
- Since we don't have time to document every thing , Minimum documentation is needed. More emphasis is given on the **working software** with teams of high expert

…Development methodologies

❑Here is a graphical illustration of the **Agile Model**



- Iterative approach is taken and **working software** build is delivered **after each iteration**.

When to use the Agile Model?

- When frequent changes are required.
- When a highly qualified and experienced team is available.
- When a customer is ready to have a meeting with a software team all the time.
- When project size is small.

… System Development methodologies

Advantage(Pros) of Agile Method:

- Frequent Delivery
- Face-to-Face Communication with clients.
- Efficient design and fulfils the business requirement.
- Anytime changes are acceptable.
- It reduces total development time.

… System Development methodologies

Disadvantages(Cons) of Agile Model:

- Due to the shortage of formal documents, it creates confusion and crucial decisions taken throughout various phases can be misinterpreted at any time by different team members.
- Due to the lack of proper documentation, once the project completes and the developers allotted to another project, maintenance of the finished project can become a difficulty.
- Difficult to Handel long term project

… System Development methodologies

Agile Method

The main goal of Agile model
Customer satisfaction by delivering
the working piece of the software
within very short span of time

…Development methodologies

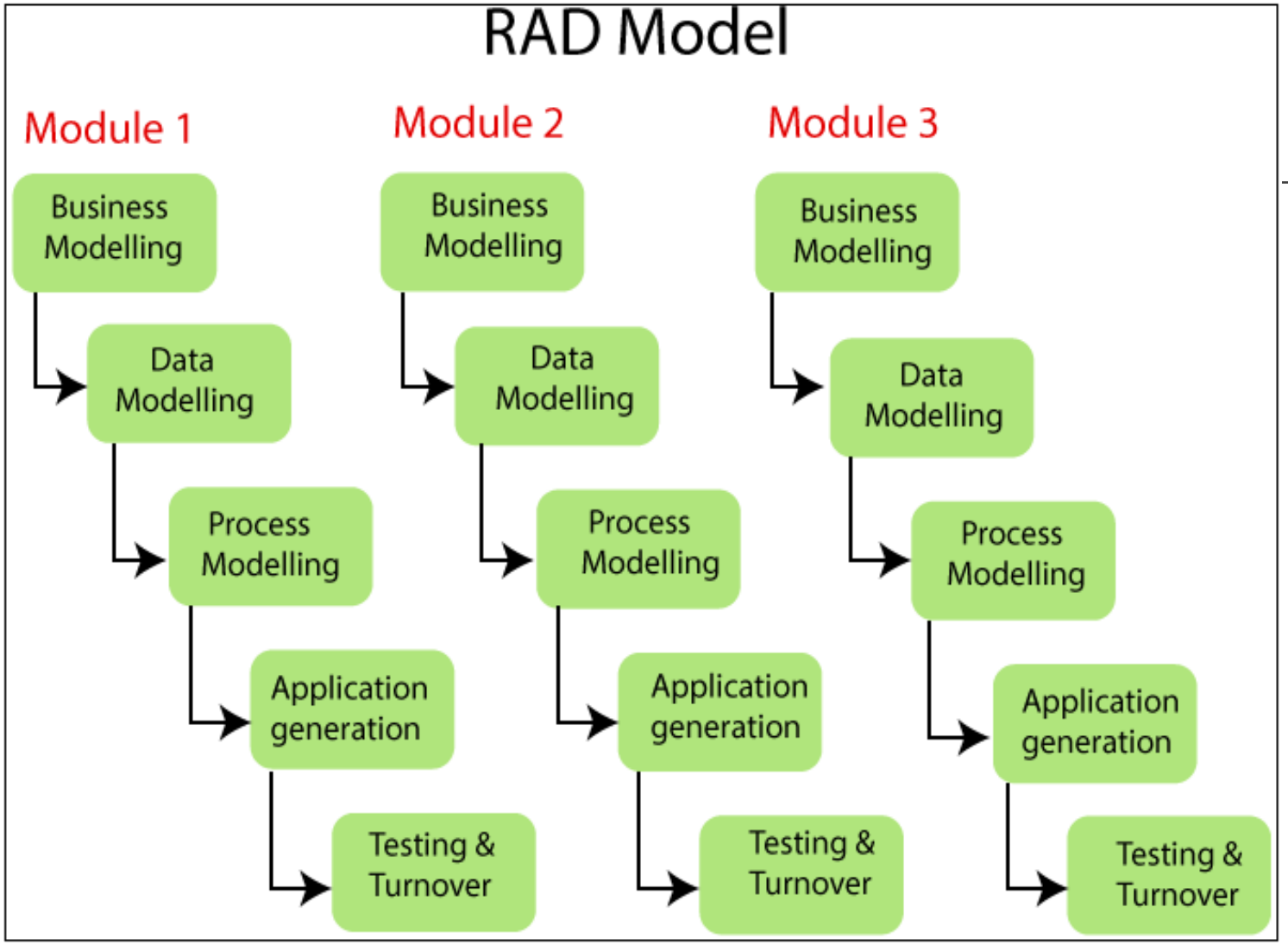
RAD (Rapid Application Development)

- It is based on **prototyping** and **iterative/incremental development** with no specific planning involved.
- It uses **minimal planning** in favor of rapid prototyping.
- The process of RAD model is building the Rapid prototype and deliver it to the clients and taking the reviews from them. If customer is satisfied then SRS document is created and designing phase is start.

….Development methodologies

- This model have small teams in encompassing developers, domain experts, customer representatives and other IT resources working progressively on their component or prototype.
- The entire project is divided into various small modules and each module is allocated to different teams to finish the working of the small modules. After that, all small modules are combined together to obtain the final project.
- **Short development cycle** (2-3 months)

…Sytstem Development methodologies



Allows to develop modules simultaneously and combined as a complete product.

…Development methodologies

Process of RAD Model :- There are **four phases** in this model :-

1-Define the Requirements:-

- At the very beginning, rapid application development sets itself apart from traditional software development models. It doesn't require as to sit with end users and get a detailed list of specifications; instead, **it asks for a broad requirement**. The broad nature of the requirements helps us take the time to segment specific requirements at different points of the development cycle.

2- Prototype:-

- In this phase, developer evaluates the customer satisfaction by delivering the prototype and taking the reviews from them. If the customer is satisfied then developer starts implementation.

…Development methodologies

3- Construction :-

- Prototype is refining and all the modification ,correction and improvements is done in this phase. This phase helps us to convert the process and modules into the final working product.

4- Deployment :-

- This is the last stage of the RAD model. In this phase, all the independent modules are evaluated separately .The tools and sub-parts of product makes the testing of the product very easy.

…Development methodologies

Advantages of RAD Model :-

- RAD model completes the project in a short period of time.
- The progress and development of project can be check on various stages .
- This model uses the powerful techniques and tools.
- Prototype is delivered to the customer so the customer is satisfied.
- It has more flexibility and adaptability to acquire the new requirements.
- Reusability of the components is increased.

Disadvantages of RAD Model :-

- Team leader must to do work with developers to complete the work on time.
- Customer involvement are needed .
- There are no reusable component are used to lead the failure of the project
- Requirements should be are clearly specified.
- It can be more complex if prototype is refined again and again.
- RAD model is not suitable for the short projects.

Summery- RAD vs. Waterfall vs. Agile Models

RAD	Waterfall	Agile
Builds a functional, working model of the application in the fastest way possible	Emphasizes intensive planning and follows through on set objectives	Builds the app by breaking down large objectives into smaller 'sprints'
Perfect for projects that require the shortest time to complete	Projects are thoroughly planned and execution is typically time-consuming	Helps develop projects in periodical milestones or 'sprints'
Can adjust to changing requirements	Does not allow for changes once planning is done	Can quite easily adjust to changes even at later stages
Involves clients throughout the development cycle	Only involve clients during the planning stage	Involves clients throughout the development
Prioritizes functionality over aspects of UI/UX	Considers all aspects of the app before deployment	UI/UX takes as much priority as functionality

Questions

Q1: Where is iterative model used?

Q2: Is waterfall model iterative?

Q3: What are the limitations of the iterative model?

Q4. What is the difference b/n Waterfoall and V-Model

END OF CHAPTER 1