

B.Tech Software Engineering

KCS-601

With
Notes

UNIT-1

Introduction
(Software Engineering)
(in one video)

AKTU Exam

Topics to be covered...

Introduction to Software Engineering

Why is Software Engineering required?

Importance of Software Engineering

Program vs. Software

Software Crisis

SDLC Models

Waterfall model

Prototype model

Iterative Waterfall model

Spiral model

Evolutionary development model

Iterative enhancement model

Product vs Process

Most Important Questions

Happy Ending!

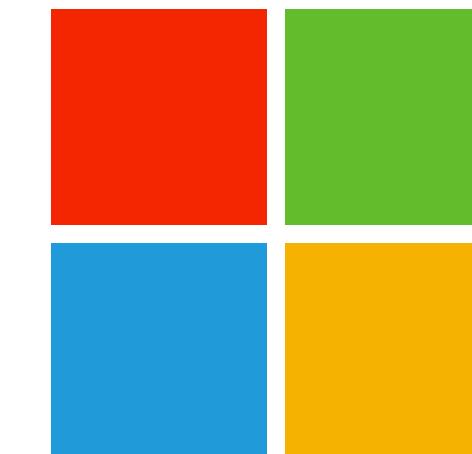


Introduction to Software Engineering

Software Engineering

Software + Engineering

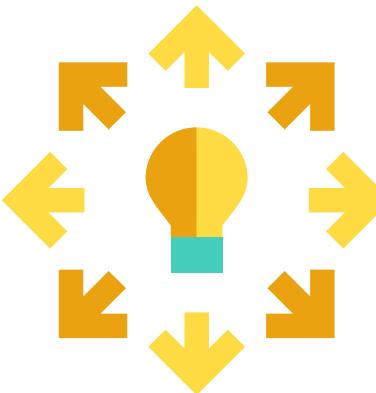
Software Engineering is an engineering branch related to the evolution of software product using well-defined scientific principles, techniques, and procedures. The result of software engineering is an effective and reliable software product.



Why is Software
Engineering required?

Software Engineering required....

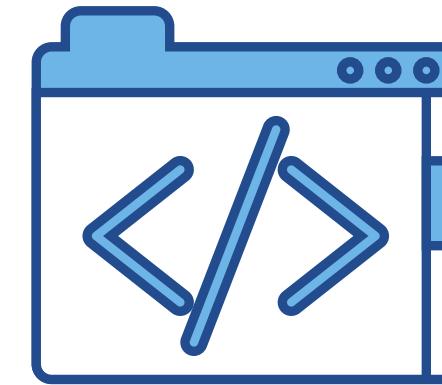
- To manage Large software
- For more Scalability
- Cost Management
- To manage the dynamic nature of software
- For better quality Management



Importance of Software Engineering

Importance of Software Engineering

- Reduces Complexity
- Minimize Software cost
- Decrease time
- Handling Big Project
- Effectiveness
- Reliable software



Program **vs** Software

Program vs. Software

Program:

Program is a combination of source code & object code.



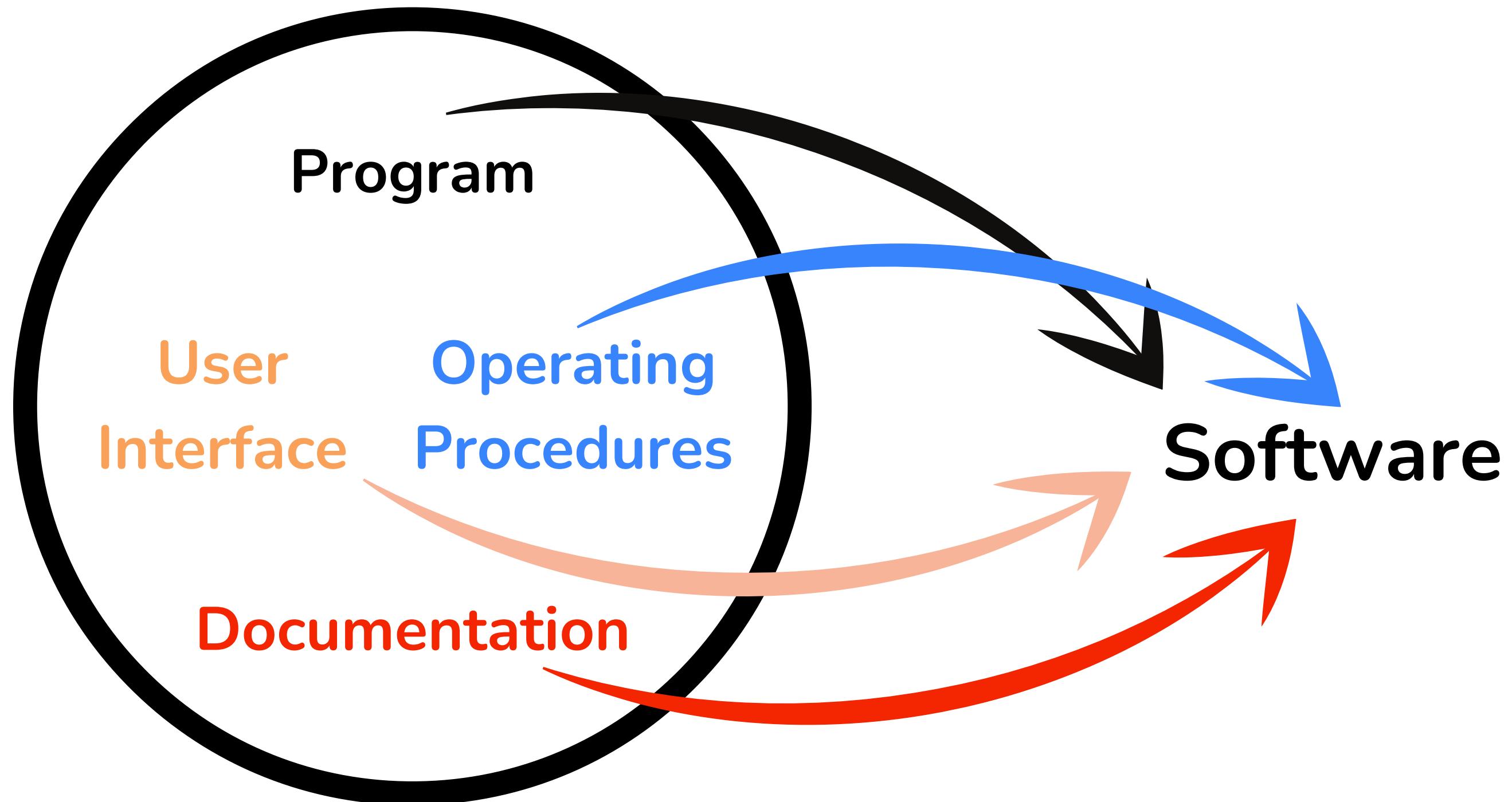
```
if ($window.scrollTop() > header1_initialDistance) {  
    header1.css('padding-top', '' + $window.scrollTop() + 'px');  
}  
else {  
    header1.css('padding-top', '' + header1_initialPadding + 'px');  
}  
  
if ($window.scrollTop() > header2_initialDistance) {  
    if (parseInt(header2.css('padding-top'), 10) > header2_initialPadding) {  
        header2.css('padding-top', '' + $window.scrollTop() - header2_initialPadding + 'px');  
    }  
}  
else {  
    header2.css('padding-top', '' + header2_initialPadding + 'px');  
}
```

Software:

Software is a set of instructions, data or programs used to operate computers and execute specific tasks.



Software



Engineering in One Video (EIOV)

Watch video on  YouTube

Software Crisis

Software Crisis

Software crisis is a set of difficulties or problems encountered while developing software.

1. **Size:** Software is becoming more expensive and more complex with the growing complexity and expectation out of software. For example, the code in the consumer product is doubling every couple of years.
2. **Quality:** Many software products have poor quality, i.e., the software products defects after putting into use due to ineffective testing technique. For example, Software testing typically finds 25 errors per 1000 lines of code.

Software Crisis

1. **Size**
2. **Quality**
3. **Cost:** Software development is costly i.e. in terms of time taken to develop and the money involved. For example, Development of the FAA's Advanced Automation System cost over \$700 per lines of code.
4. **Delayed Delivery:** Serious schedule overruns are common. Very often the software takes longer than the estimated time to develop, which in turn leads to cost shooting up. For example, one in four large-scale development projects is never completed.



Engineering

IN ONE VIDEO

CONNECT, LEARN AND GROW



Engineering in One Video (EIOV)

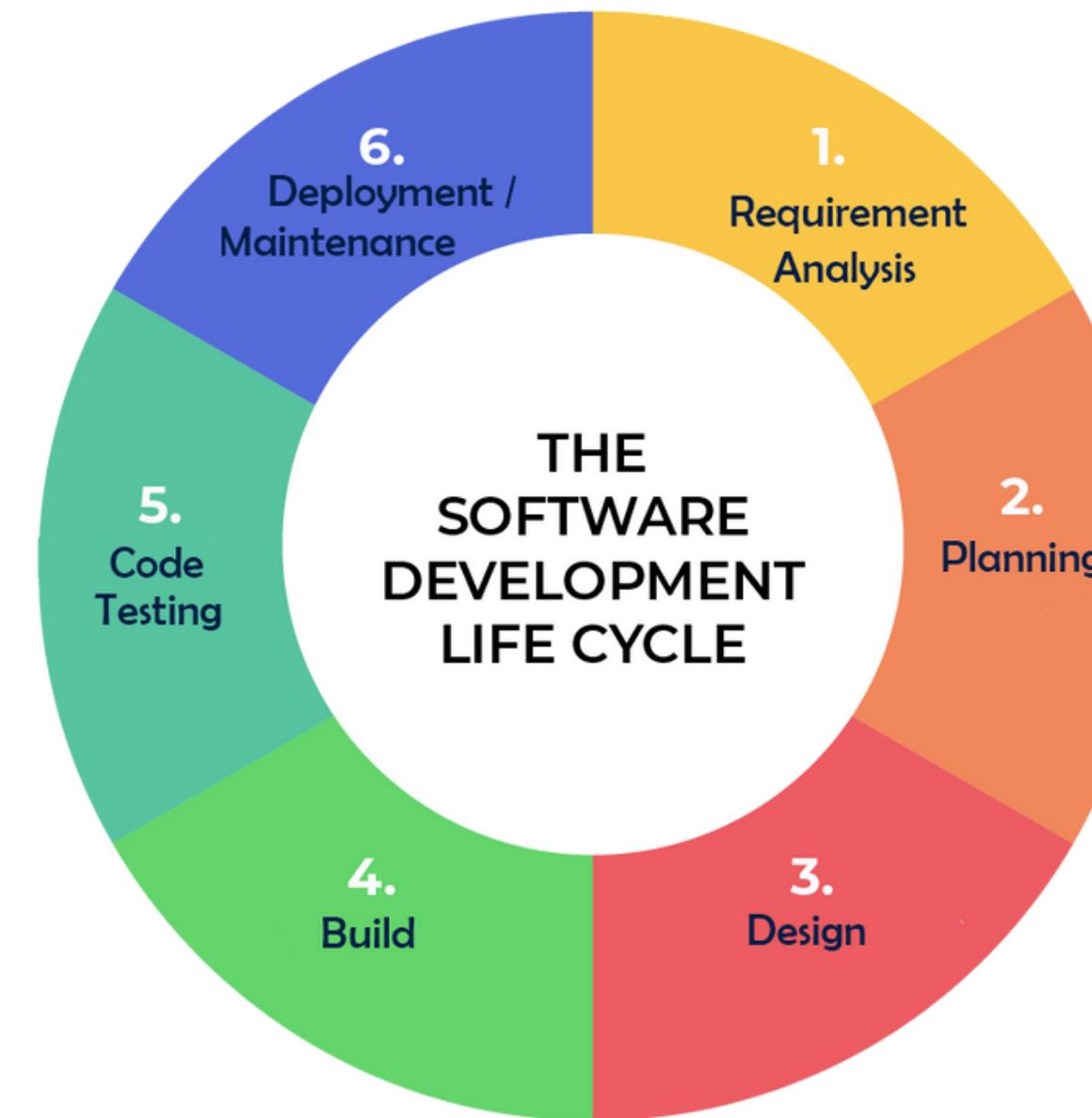
Watch video on  YouTube

SDLC Models

(Software Development Life Cycle)

SDLC Models

1. A software Development life cycle model is a pictorial and diagrammatic representation of the software life cycle.
2. SDLC is used by the software industry to design, develop and test high quality softwares.



SDLC Models

Phase 1: Requirement Analysis:

- Gathering business requirement
- Creating process diagrams
- Performing a detailed analysis

Phase 2: Planning

- Identification of the system for development
- Feasibility assessment
- Creation of project plan

Phase 3: Design

- This phase includes business rules, pseudo-code, screen layouts, and other necessary documentation.
- Designing of software model.

SDLC Models

Phase 4: Build/Development:

- The implementation of design begins concerning writing code. Developers have to follow the coding guidelines described by their management and programming tools like compilers, interpreters, debuggers, etc. are used to develop and implement the code.

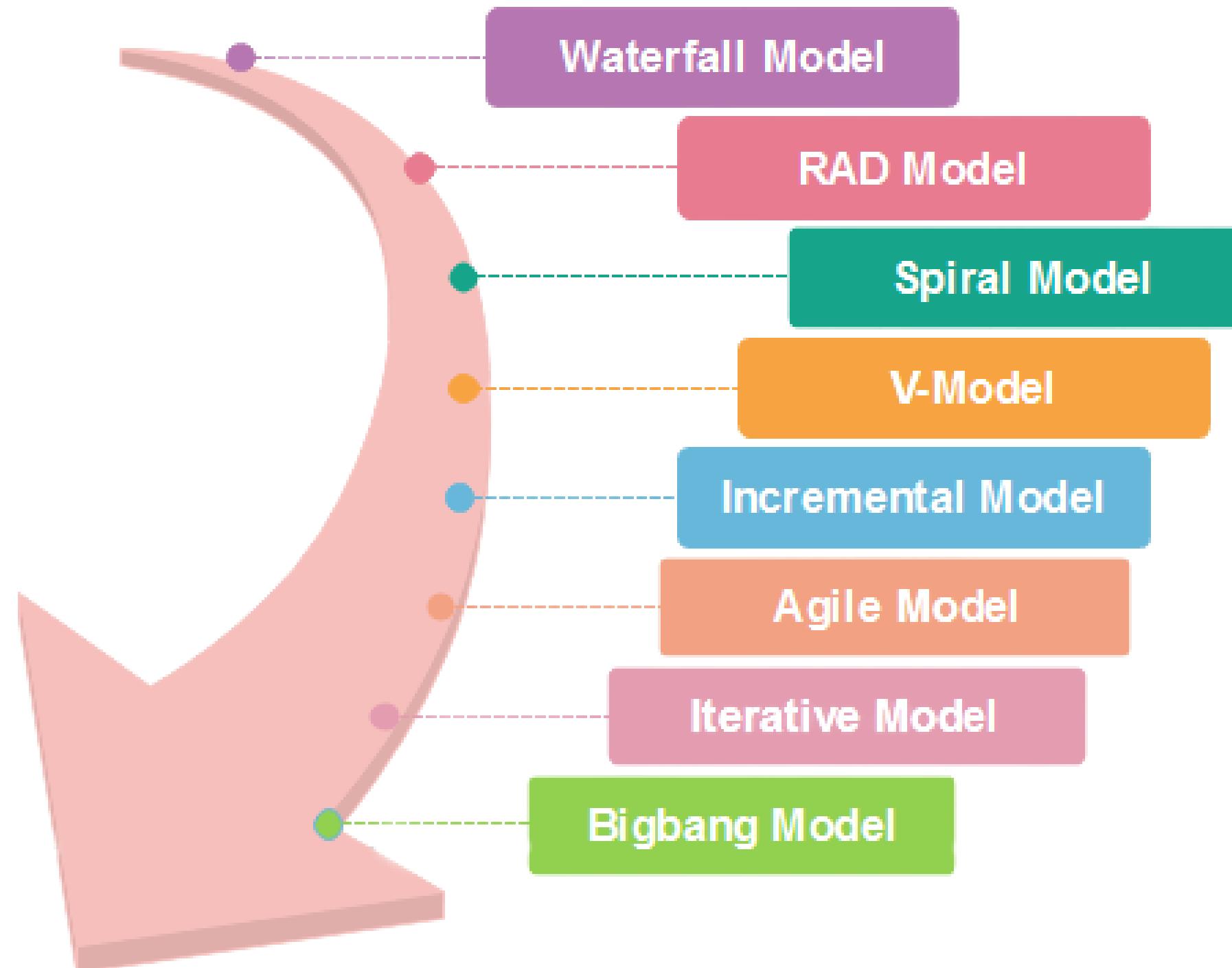
Phase 5: Code Testing

- Writing test cases
- Execution of test cases
- Tester follows the Software Testing Life Cycle activities to check the software for errors, bugs, and defects.

Phase 6: Deployment/Maintenance

- In Deployment software is deployed to a real-life environment where the actual user begins to operate the software.
- Support the software users.
- Software maintenance.
- Software changes and adjustment

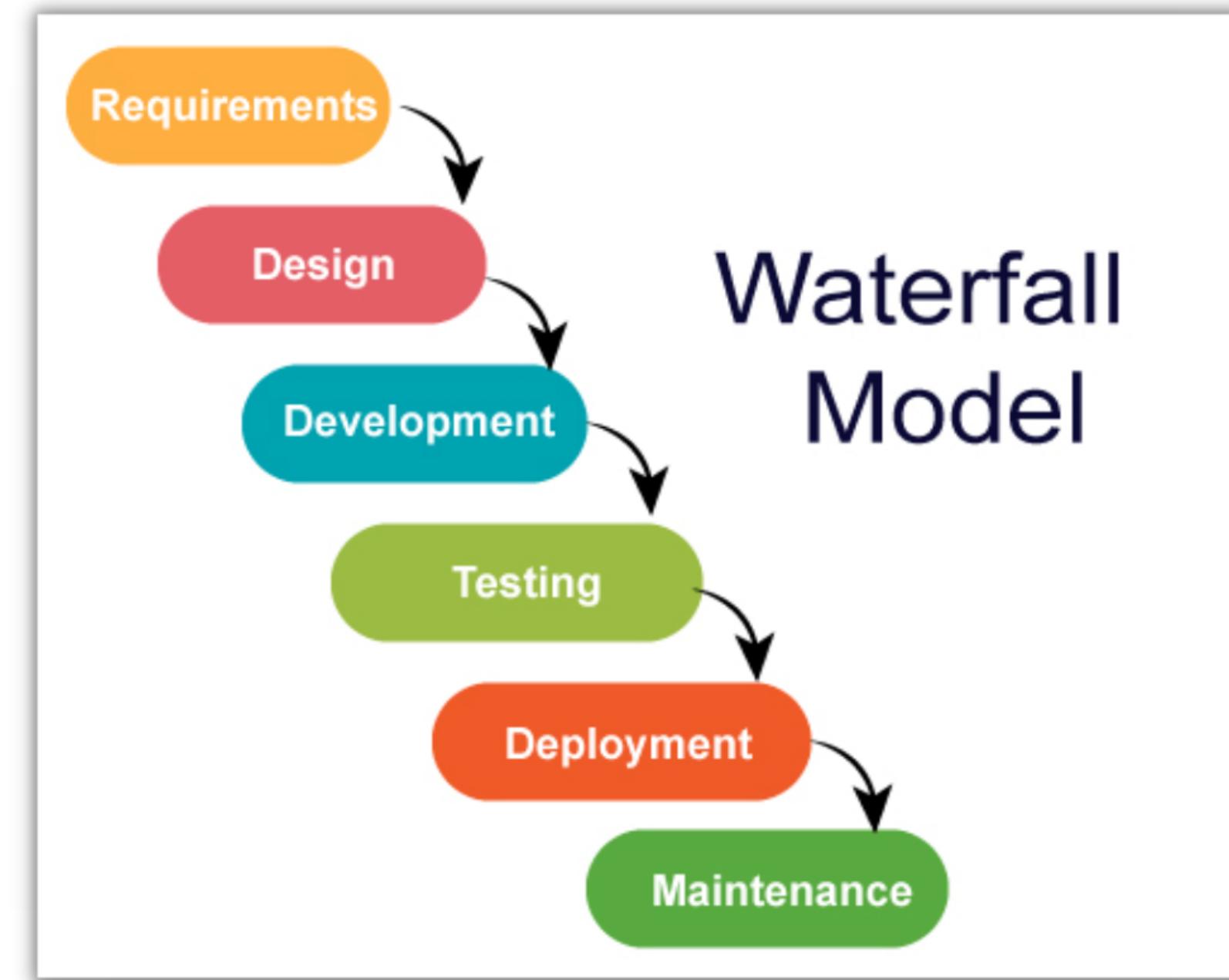
SDLC Models



Waterfall model

Waterfall model

In Waterfall Model progress is seen as flowing steadily downwards (like a waterfall).



Waterfall model advantages & Disadvantages

Advantages:

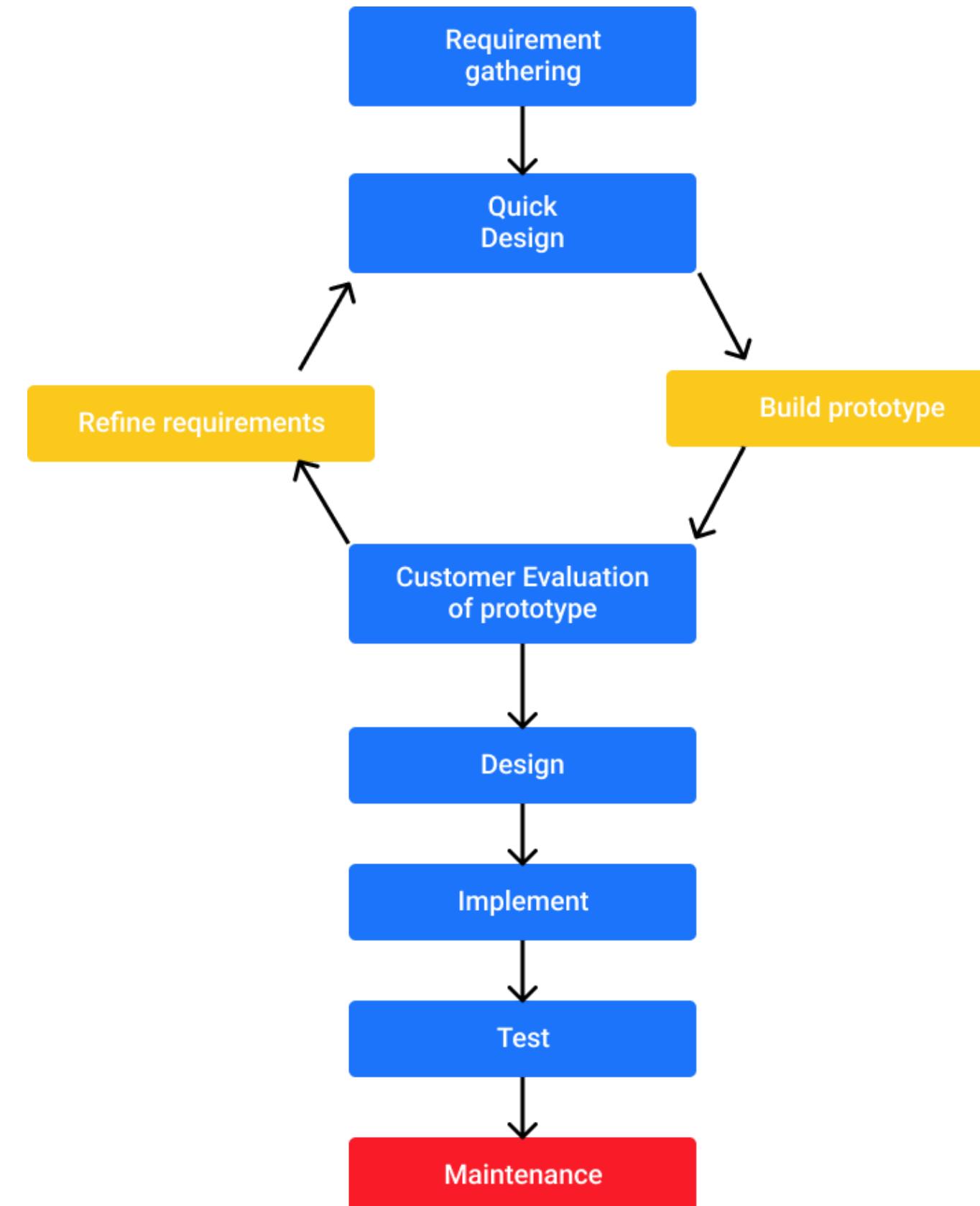
- it is easy to use and understand.
- No overlapping between the phases.
- This model is suitable for small projects.
- Provides better documentation for the employees.
- Low Cost
- It is easy to maintain.

Disadvantages:

- This model does not work for large projects.
- We can't go back at previous phase to change anything or any requirement.
- If customer is not satisfied with the project then it is hard to change the requirements.
- High risk to make a project.

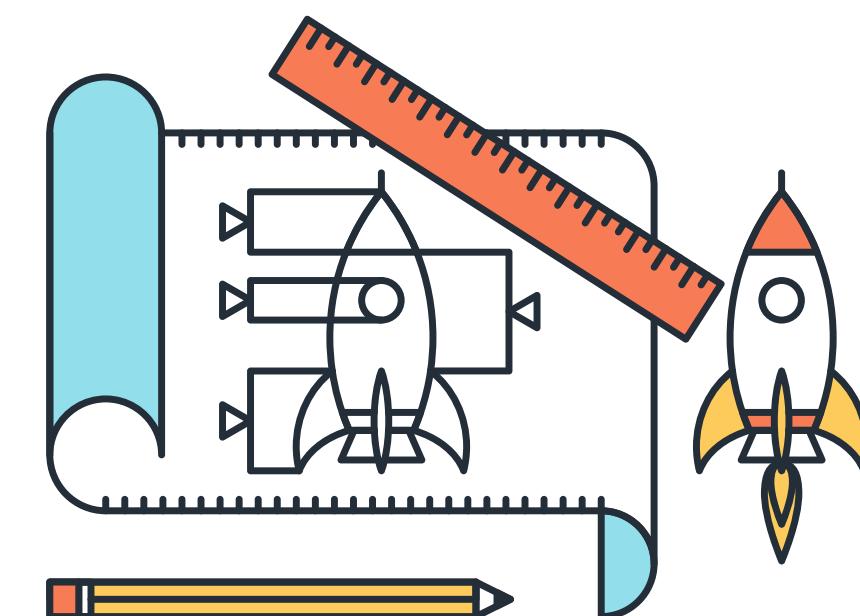
Prototype model

Prototype model



Prototype model

The prototype model requires that before carrying out the development of actual software, a working prototype of the system should be built. A prototype is a toy implementation of the system. A prototype usually turns out to be a very crude version of the actual system, possibly exhibiting limited functional capabilities, low reliability, and inefficient performance as compared to actual software. In many instances, the client only has a general view of what is expected from the software product.



Prototype model advantages & Disadvantages

Advantages:

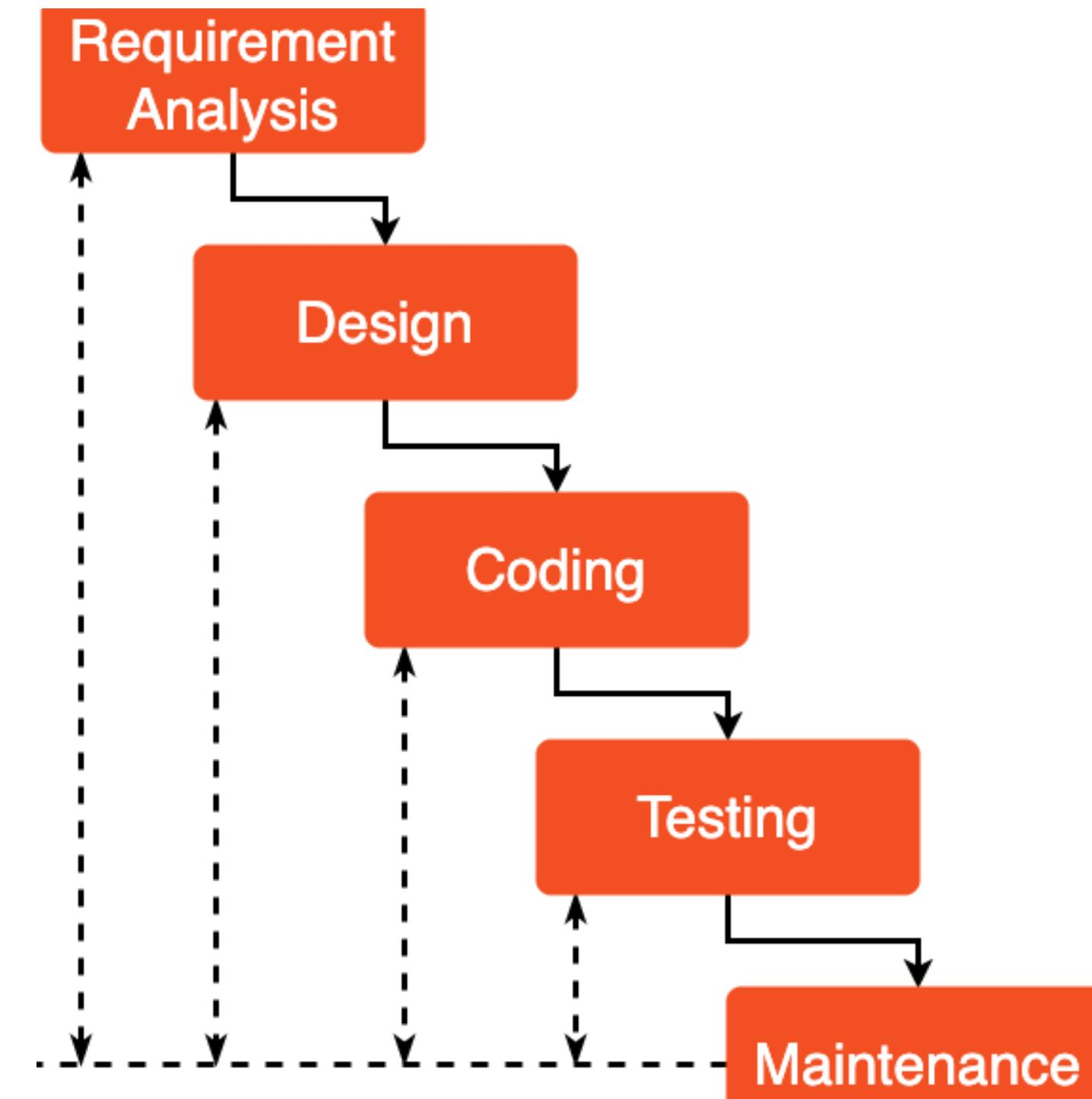
- It provides higher customer satisfaction and get reviews from the customers.
- Requirement can change easily according to the customer. It means this model has high adaptability.
- It gives high flexibility.
- Prototype model increase the involvement of customer.
- Reduce Maintenance cost.

Disadvantages:

- Difficult to know how long the project will last.
- Prototyping tools are expensive.
- If a customer refuses to accept the prototype, then there may be wastage of a lot of cost and developer's efforts.
- It is a time-consuming process.

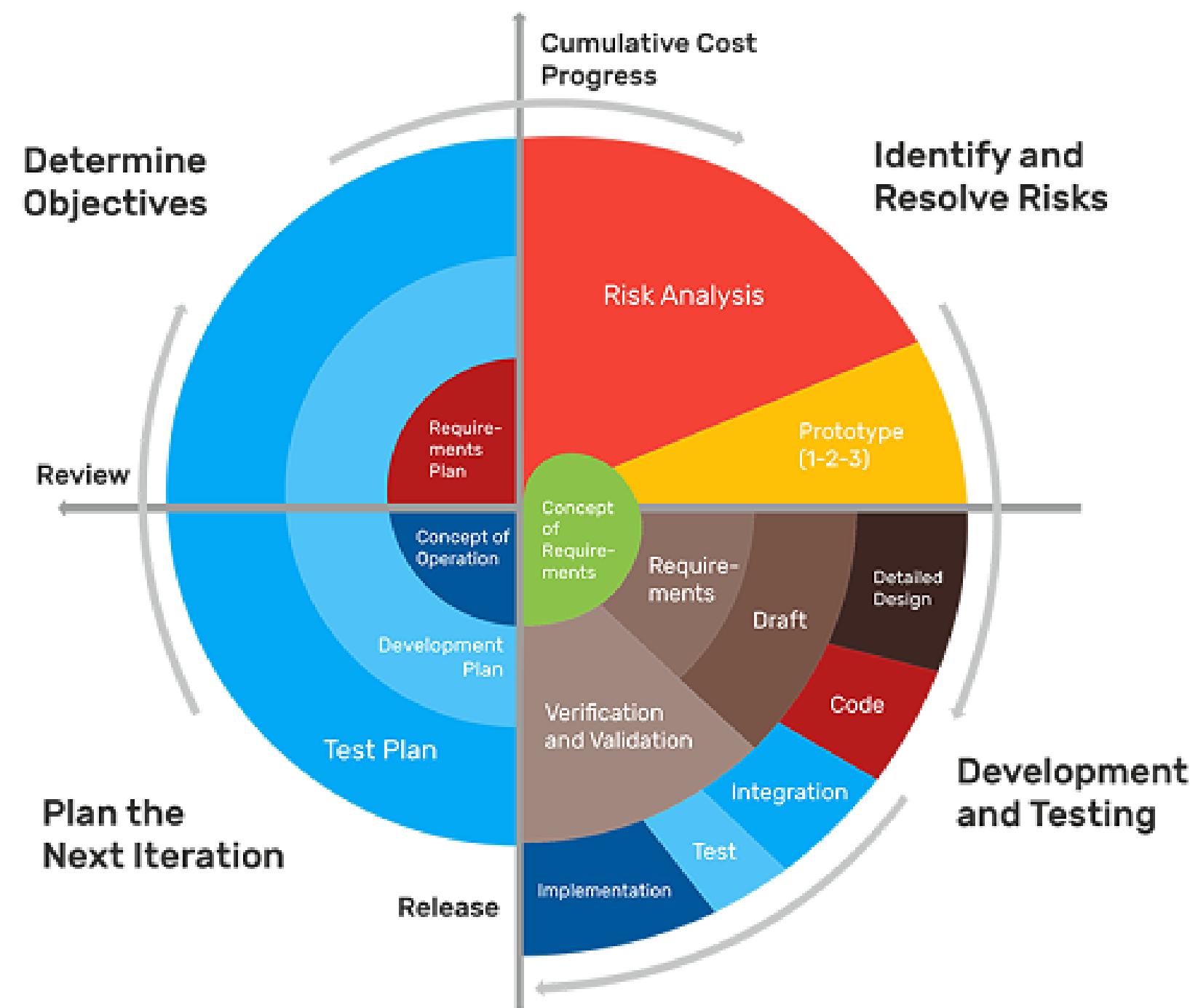
Iterative Waterfall model

Iterative Waterfall model



Spiral model

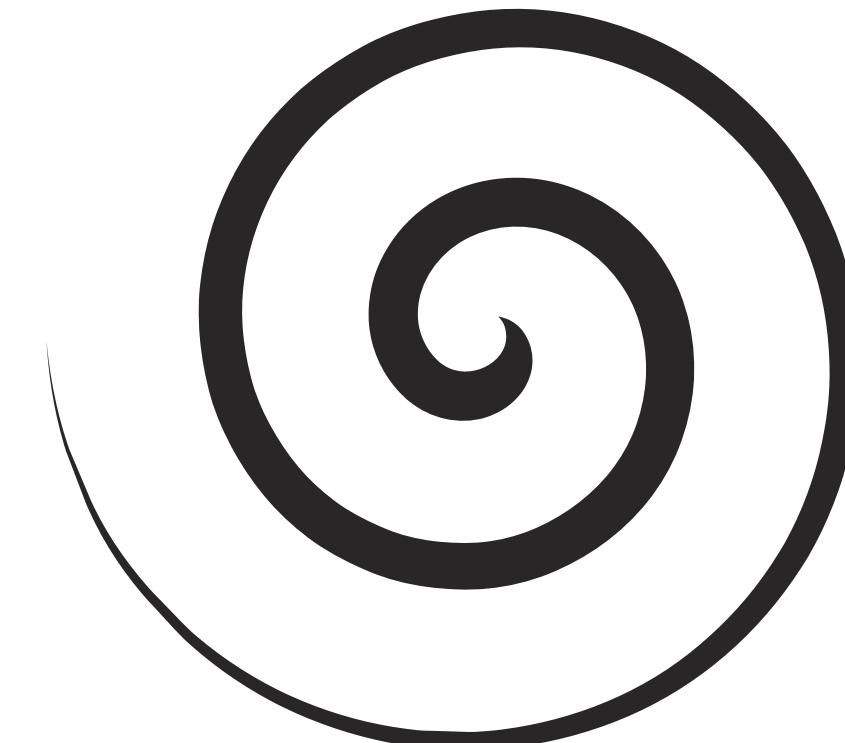
Spiral model



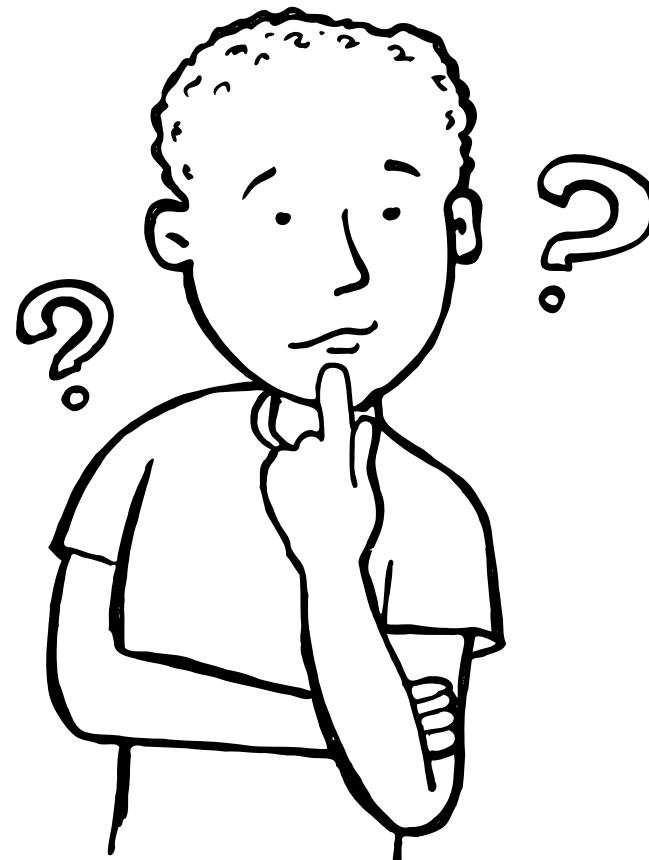
Spiral Model Methodology and its Phases

Spiral model

- The Spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model.
- A Spiral model has following 4 phases:
 - Planning phase
 - Risk analysis phase
 - Development & Testing phase
 - Plan the next iteration



When to use Spiral Model?

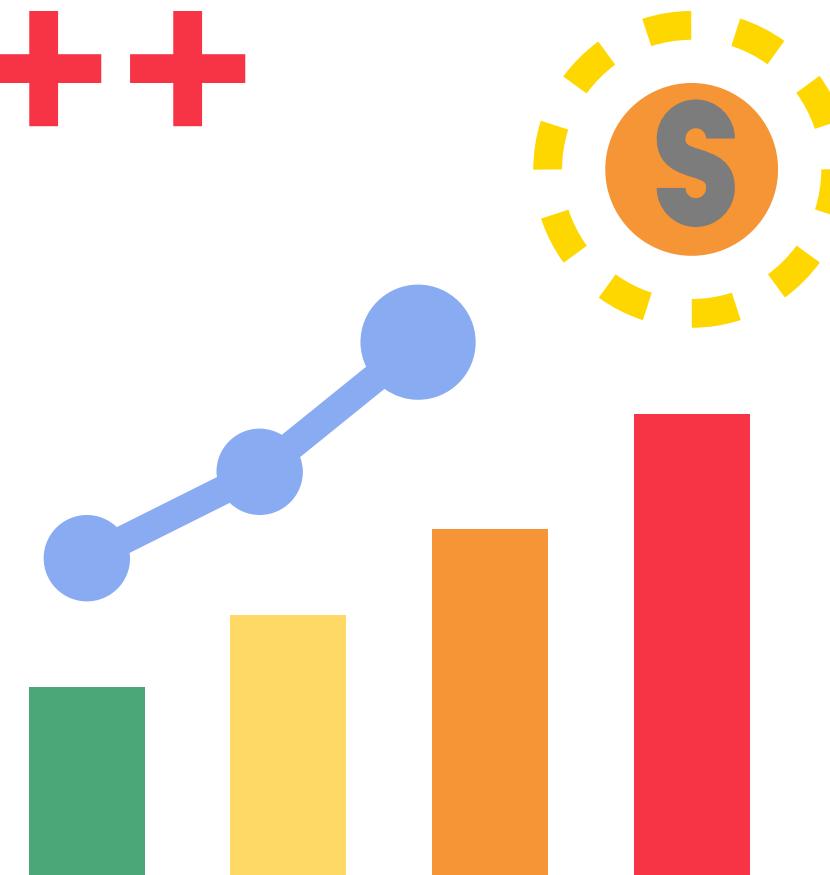


- When deliverance is required to be frequent.
- When the project is large
- When requirements are unclear and complex
- When changes may require at any time
- Large and high budget projects

Spiral model advantages & Disadvantages

Advantages:

- High amount of risk analysis.
- Useful for large and mission-critical projects.



Disadvantages:

- Can be a costly model to use.
- Risk analysis needed highly particular expertise.
- Doesn't work well for smaller projects.



Engineering

IN ONE VIDEO

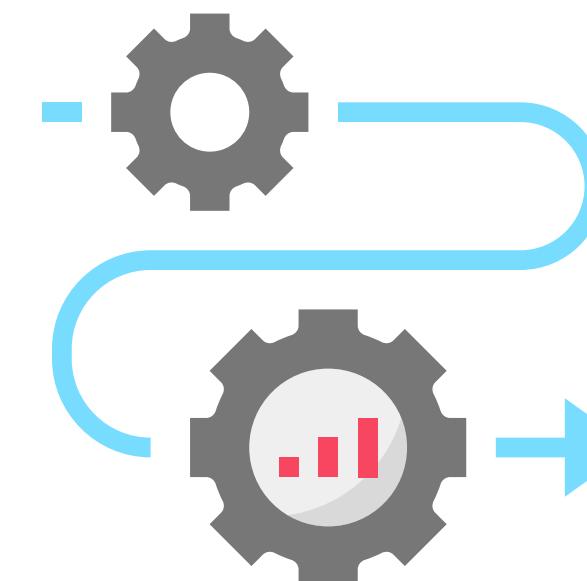
CONNECT, LEARN AND GROW



Evolutionary development model

Evolutionary development model

- Evolutionary models are iterative type models which allow to develop more complete version of the software.
- In this model, the system is first broken down into the several modules or functional units that can be incrementally implemented and delivered.
- The developer first develop the core modules of the system.
- This model is also known as the successive versions model.



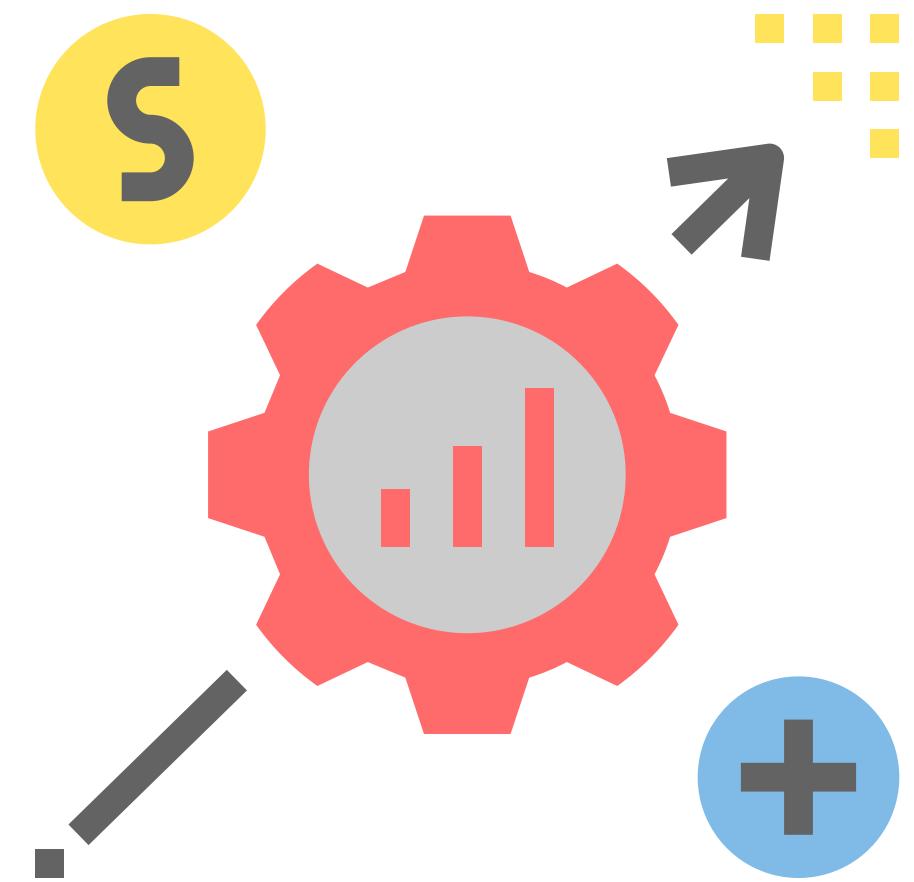
Evolutionary development model advantages & Disadvantages

Advantages:

- In evolutionary model, a user gets a chance to experiment partially developed system.
- It reduces the error because the core modules get tested thoroughly.

Disadvantages:

- Sometimes it is hard to divide the problem into several versions that would be acceptable to the customer which can be incrementally implemented and delivered.



Iterative enhancement model

Iterative enhancement model

In this Model, you can start with some of the software specifications and develop the first version of the software. After the first version if there is a need to change the software, then a new version of the software is created with a new iteration. Every release of the Iterative Model finishes in an exact and fixed period that is called iteration.



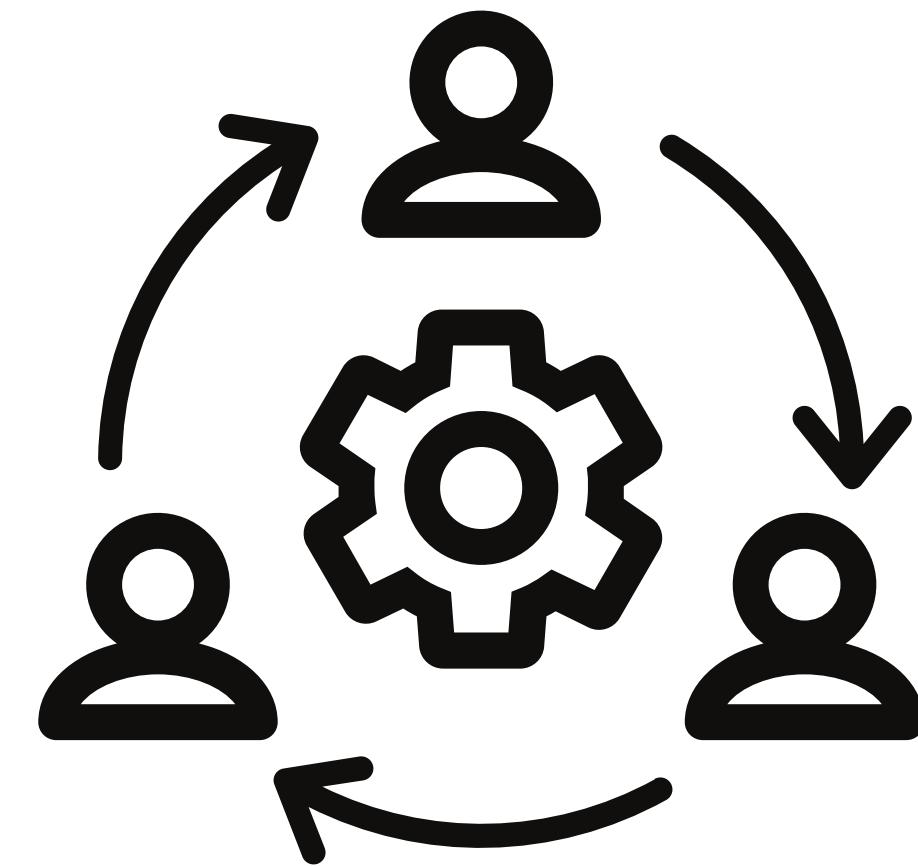
Iterative enhancement model advantages & Disadvantages

Advantages:

- Testing and debugging during smaller iteration is easy.
- A Parallel development can plan.
- Risks are identified and resolved during iteration.
- Limited time spent on documentation and extra time on designing.

Disadvantages:

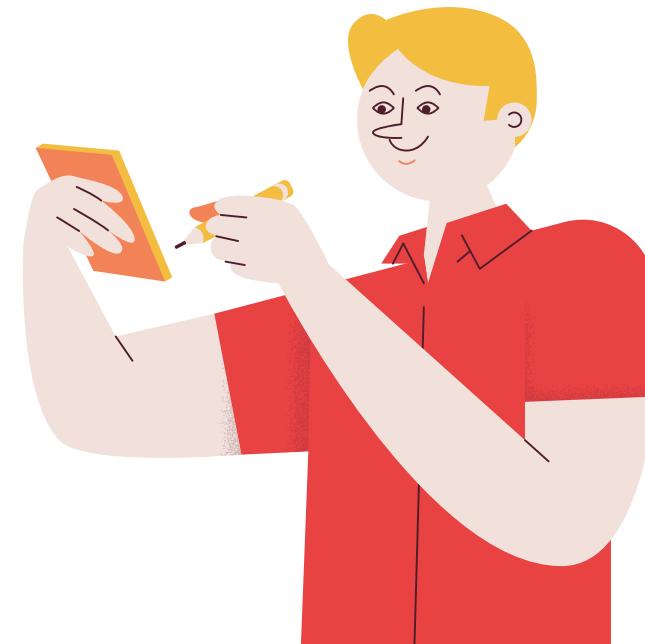
- It is not suitable for smaller projects.
- More Resources may be required.
- Requirement changes can cause over budget.
- Project completion date not confirmed because of changing requirements.



Product vs Process

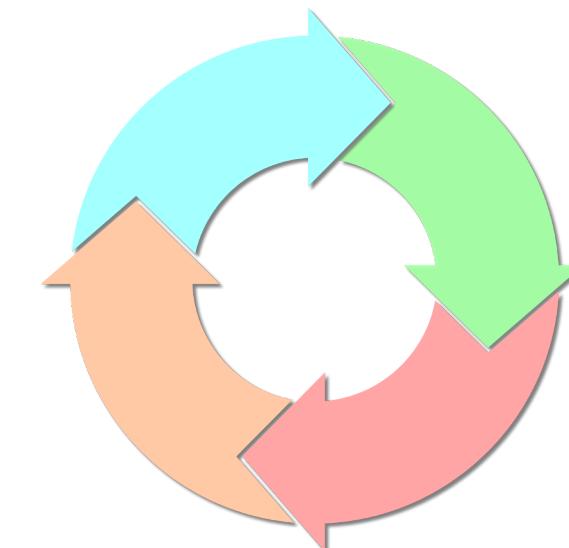
Product:

A product may be defined as an output. A product can produce the good services.



Process:

A process may be defined as an investment of time and efforts for getting a good product.



Most Important Questions

Most Important Questions

1. Explain software development life cycle. Discuss various activities during SDLC.
2. Explain iterative Waterfall and Spiral model for software life cycle and discuss activities in each phase.
3. Explain the working of Prototype model with suitable diagram.
4. Explain Waterfall model with suitable diagram.
5. What is software crises ? Discuss main reasons and results of software crises.
6. Product vs Process.
7. Define software process.
8. What are the characteristics of a software process ?

Happy Ending!



Congratulations!

