

15/10/2020
Saturday

Software Engineering: krishna ①

- Software Engineering is a systematic, disciplined, cost-effective techniques for software development.
- It is not related to only code, it is the whole approach to develop a software

Evaluation:

1945 - 65 → Origin

1965 - 85 → Crisis

1990 - 2000 → Internet

2000 - 2010 → Light weight (mobile software)

2010 - Till → AI, ML, DL (self learning software)

→ Software Engineering is a systematic approach to the analysis, implementation and maintenance of software.

→ Software Engineering is a technique which we can develop or create software for computer system and any other electronic devices.

→ In other words, SE is a process in which user needs are analyzed and software is designed based on their needs.

→ In SE, the development of software will be using well defined scientific principles, methods and processes.

→ Software Engineers build these software and applications by using designing and Programming Languages.

→ In order to create a complex S/W, we should use S/W engineering techniques as well as we should use abstraction and decomposition to reduce complexity.

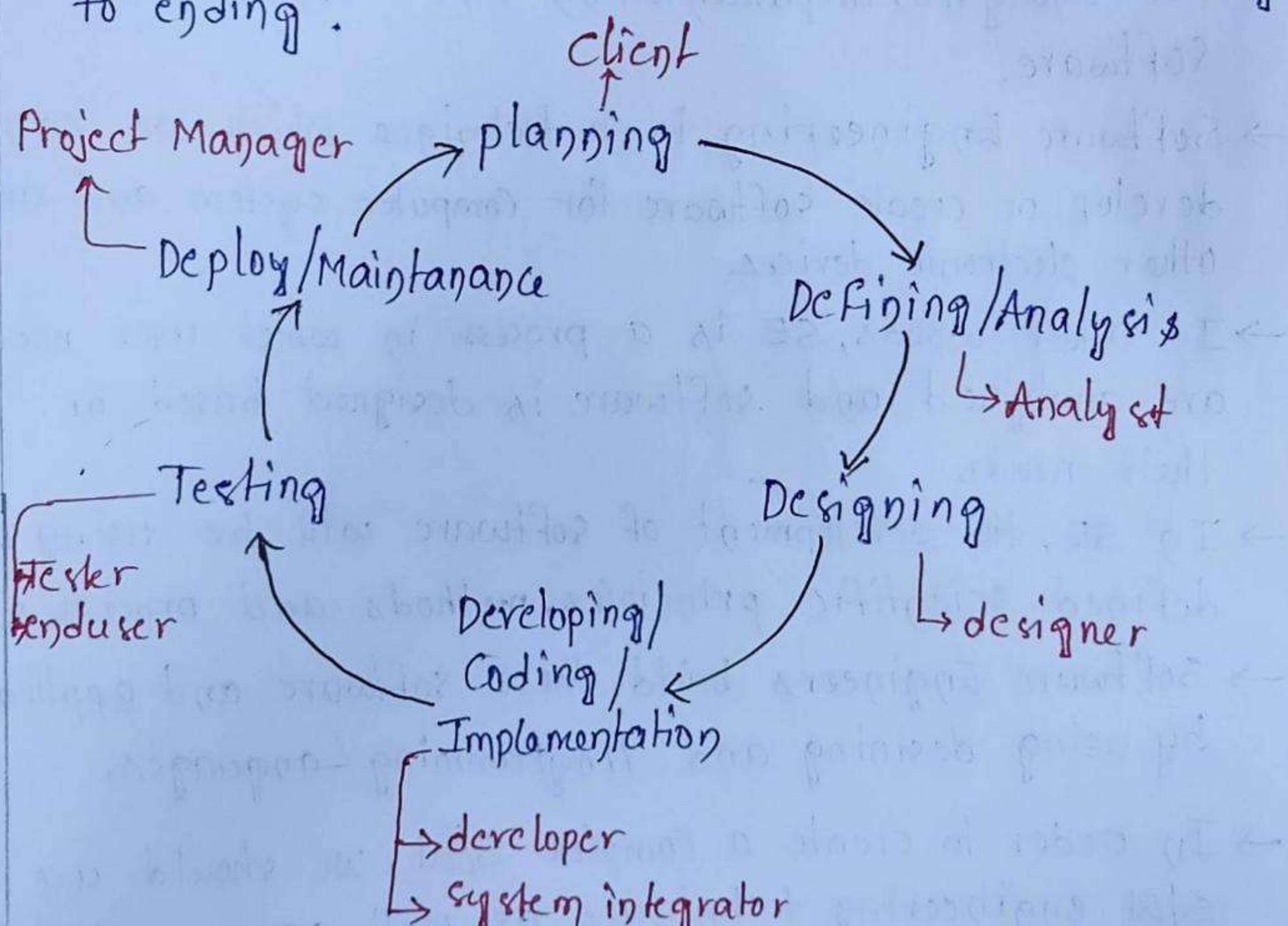
Purpose of SE:

16/10/2022
Sunday

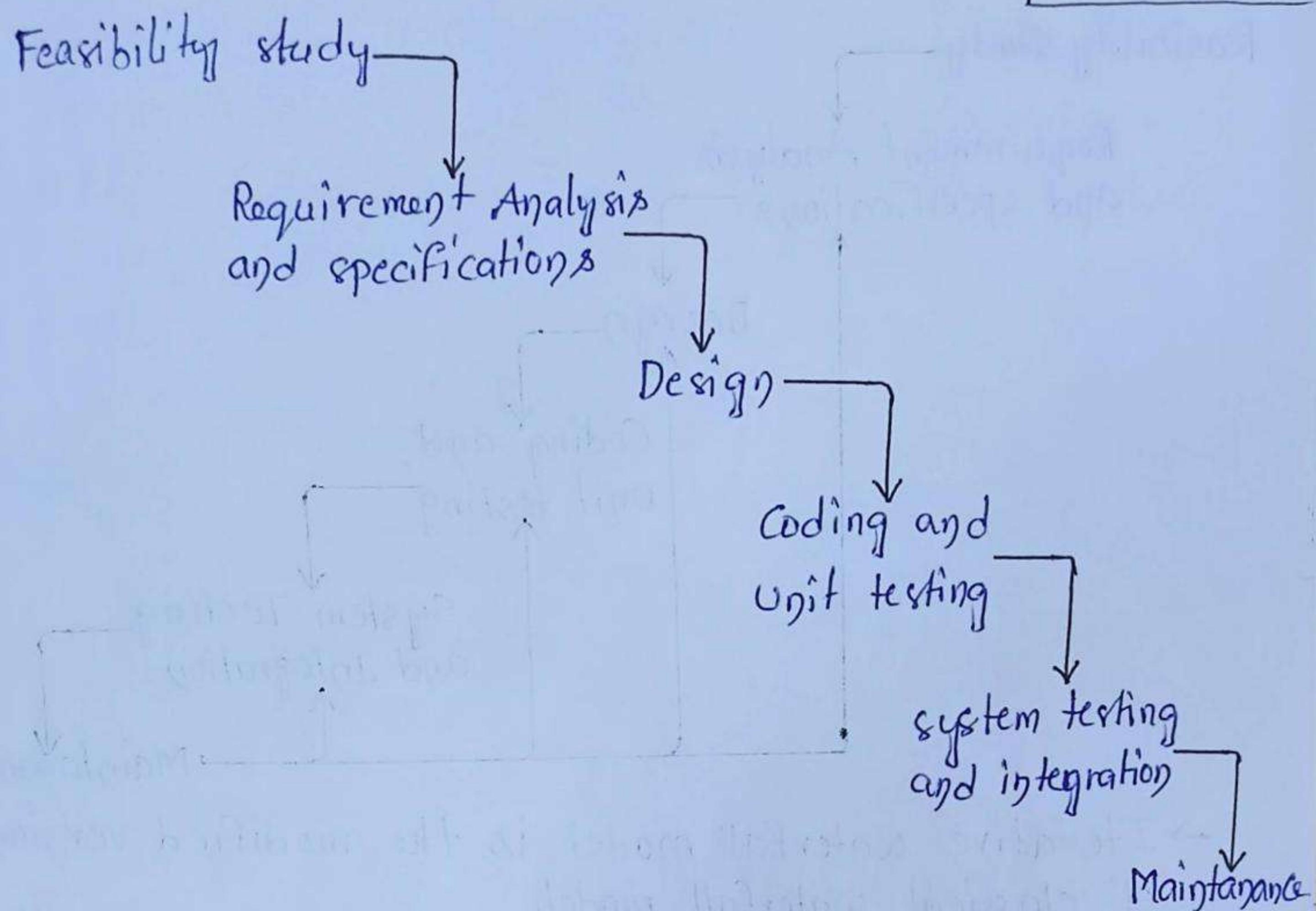
1. To manage large Software
2. For greater Scalability
3. To manage the cost
4. To manage the dynamic nature of the Software.
5. For better quality management.

SDLC:

- SDLC stands for Software Development Life Cycle model.
- It describes the sequence of phases or steps to develop any ~~project~~ software.
- SDLC means "entire lifetime of S/W from beginning to ending".



- The SDLC model is classified into various categories based on their advantages & disadvantages:
 1. Classical Waterfall Model
 2. Iterative Waterfall Model
 3. V shaped model
 4. Prototyping model
 5. Incremental Model
 6. Evolutionary Model
 7. Spiral Model



- The waterfall model is simple and classical model of all the models we have.
- This is also known as linear ~~sequential~~ sequential model.
- This model is theoretical models, not a practical model.
- In this model, each and every phase must be completed before moving to the next phase.
- This model is suitable for small projects where the technical issues are very clear.
- At the end of each phase, a review will take place.

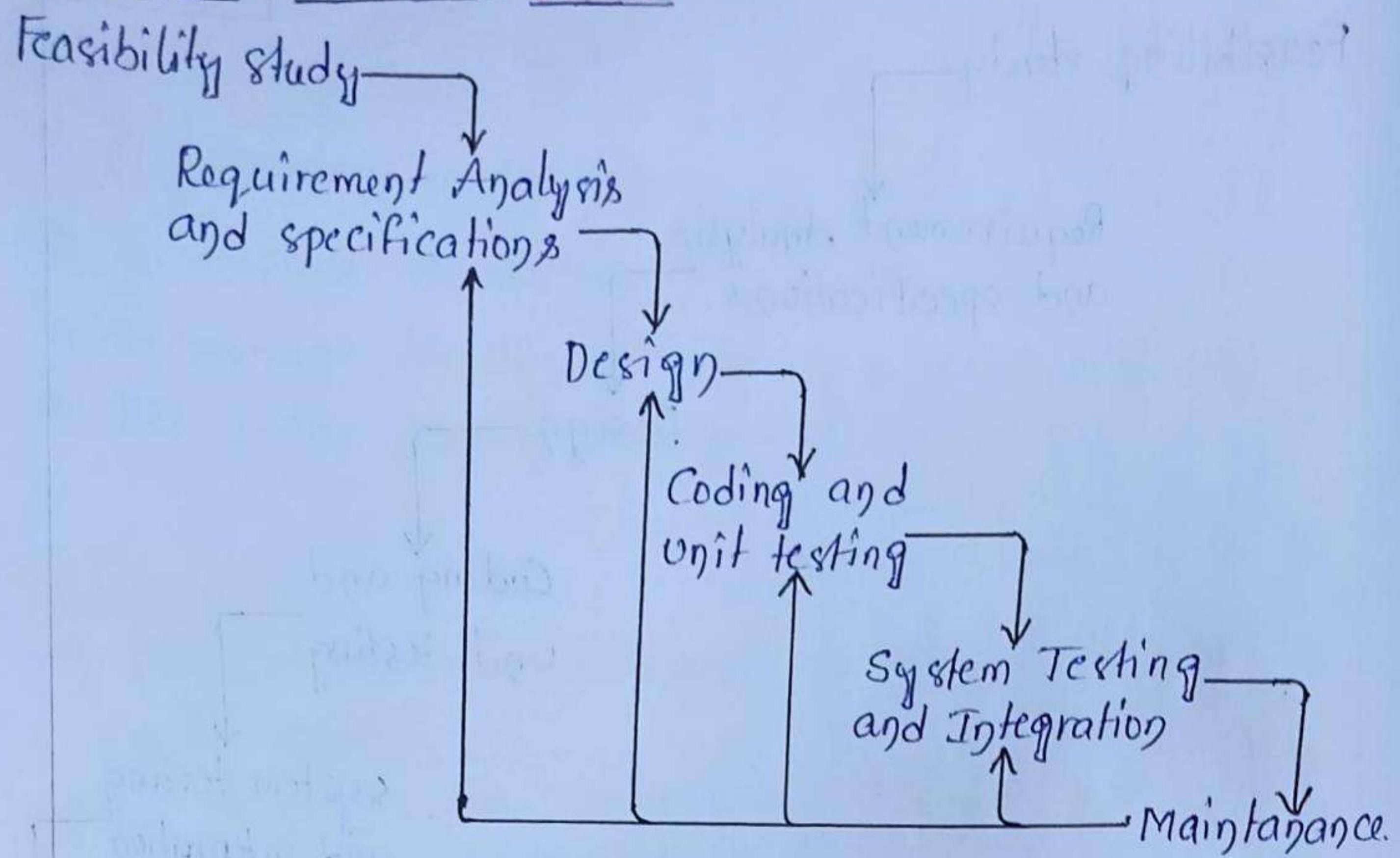
Advantages

- * Basic Model
- * Simple and easy
- * used for small projects

Disadvantages

- * No Feedback
- * No Experiment
- * No Parallelism
- * High Risk
- * 60% efforts in Maintenance.
- * Poor for Object Oriented Project.

2. Iterative Waterfall Model:



→ Iterative waterfall model is the modified version of classical waterfall model.

- * → The major modification is Feedback.
- * → There is no feedback for Feasibility study phase.

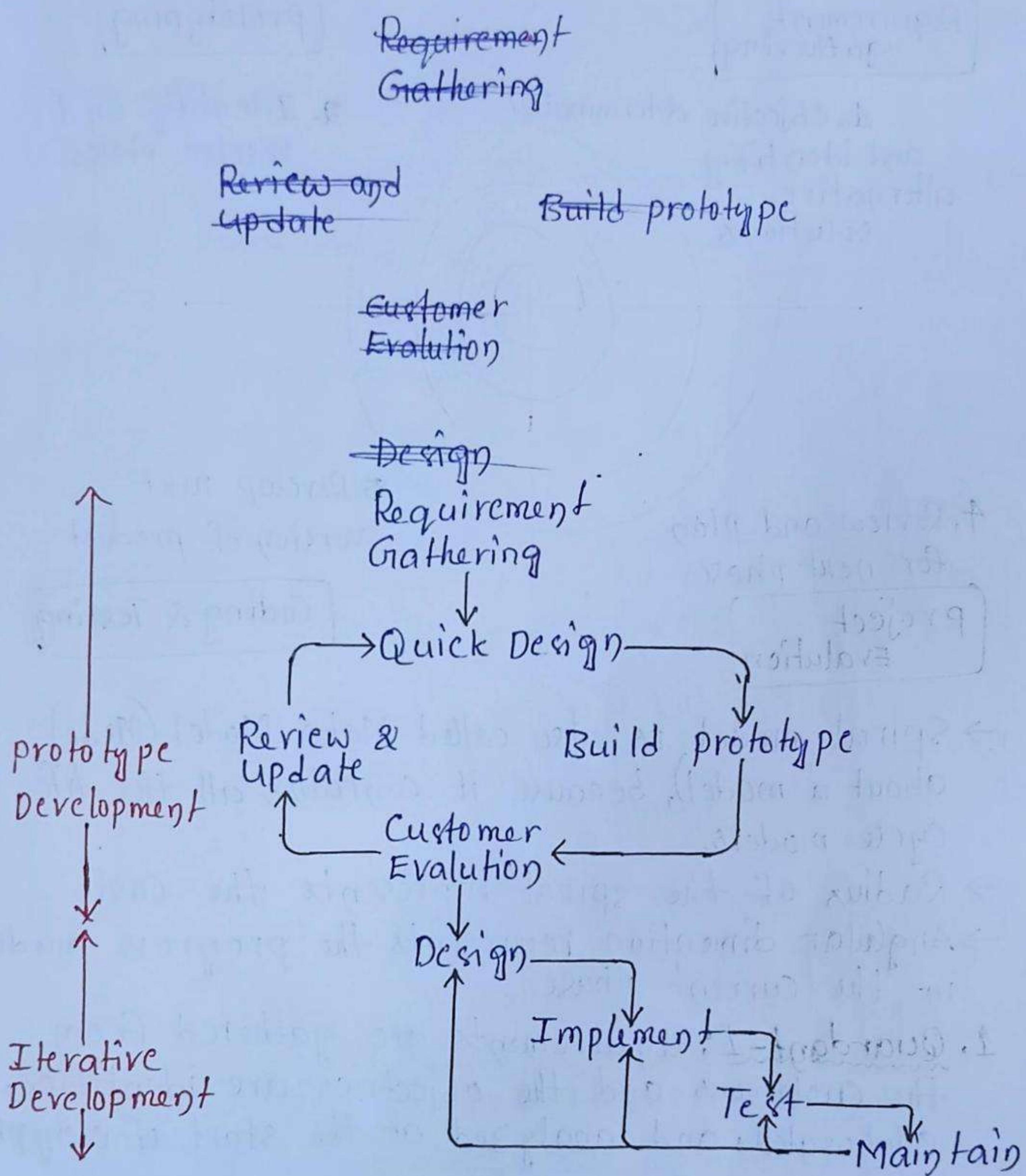
Advantages

- * Basic Model
- * Simple and easy
- * For small projects
- * Feedbacks

Disadvantages

- * No Phase Overlapping
- * No Intermediate delivery
- * Rigid (No changes)
- * Less customer Interaction

3. Prototyping Model: krishna ⑤



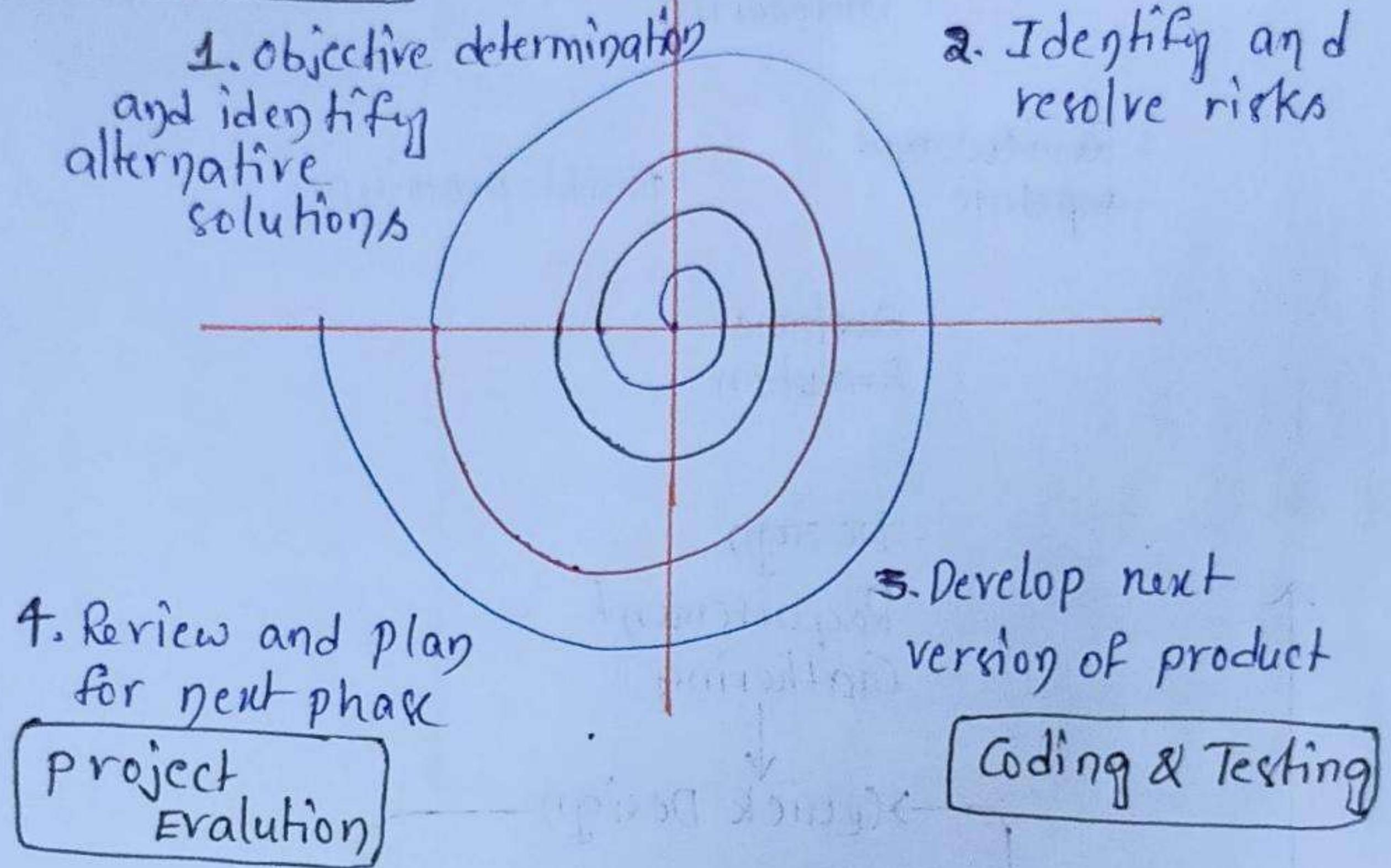
- It's a very famous model where client is involved in designing.
- Here, Prototype will be modified until the client is ~~not~~ satisfied and will go to design phase.
- It is an iterative development cycle b/w developer & client.
- It is used when customer is not clear with idea. (Advantage)
- It is referred as Throwaway model. (Advantage)
- Good for technical and requirement risks. (Advantage)
- Increase in cost of development (disadvantage)

4. Spiral Model:

krishna ⑥

Requirement gathering

prototyping



→ Spiral model is also called Meta Model (Model about a model), because it contains all the life-cycle models.

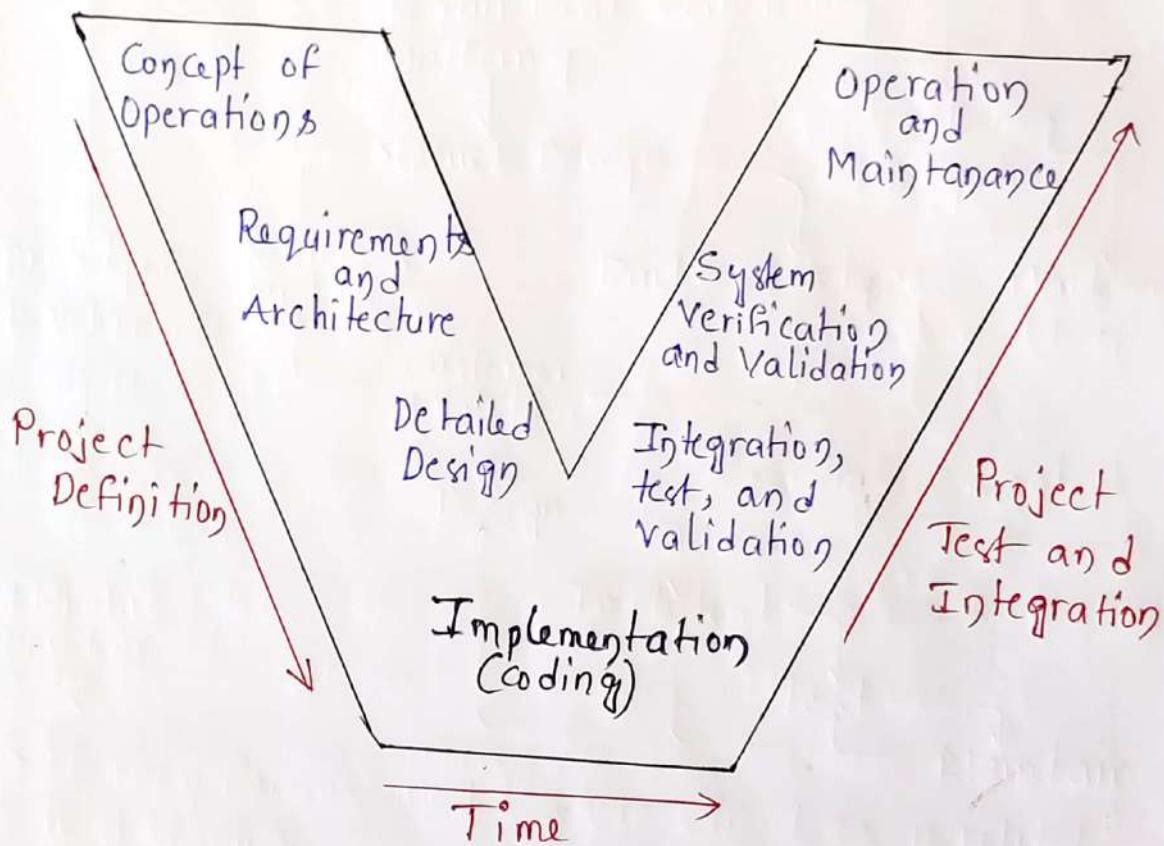
→ Radius of the spiral represents the cost!

→ Angular dimension represents the progress made in the current phase.

1. Quadrant-1: Requirements are gathered from the customers and the objectives are identified, elaborated, and analyzed at the start of every phase.

5. V-shaped Model:

- It is also known as verification & validation model
- Extension of Waterfall Model.
- Testing is associated with every phase of life cycle.
- Verification Phase: Requirement analysis, System design, Architecture design, Module design
- Validation Phase: Unit testing, Integration, System, Acceptance Testing.



Advantages

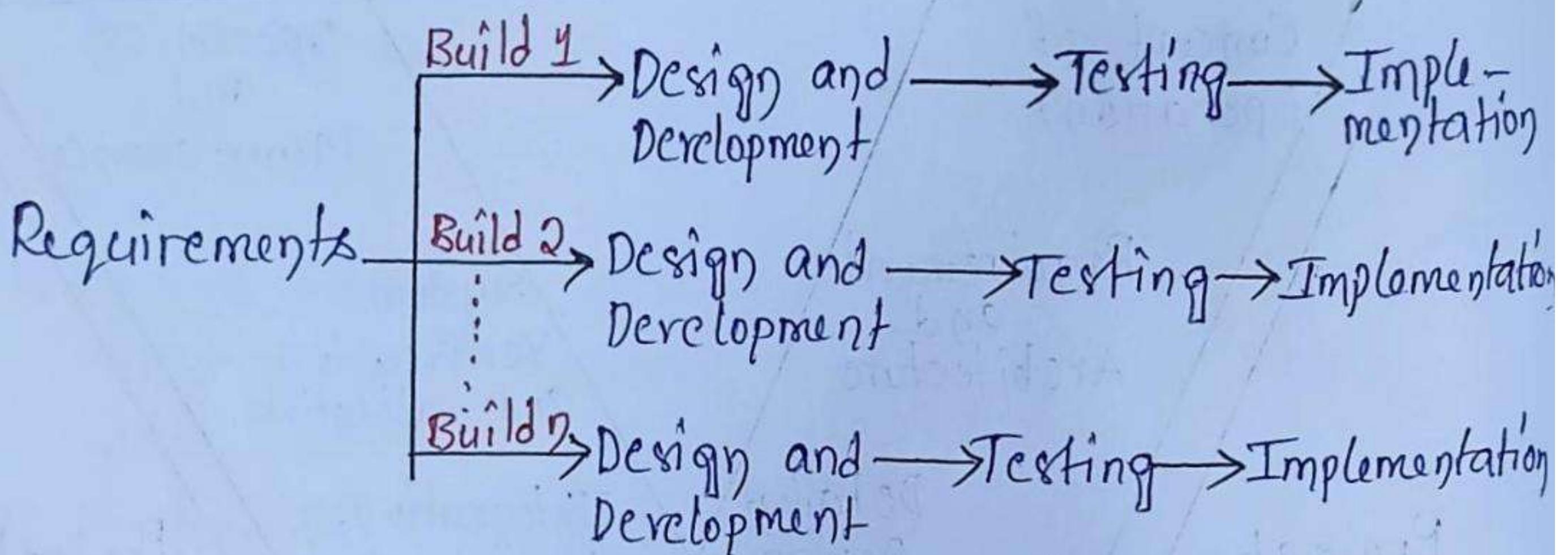
- * Time saving
- * Good understanding of project in the beginning
- * Every component must be testable.
- * Progress can be tracked easily.
- * Proactive defect tracking

Disadvantages

- * No feedback, so less no. of changes
- * Risk Analysis not done
- * Not good for big or object oriented Projects.

6. Incremental Model: krishna(8)

- Incremental model is a process of software development where requirements are broken down into multiple standalone modules of SDLC.
- Incremental development is done in steps from analysis to Implement.
- Each phases through the requirements, design, Coding and Testing, and implementation.
- Each subsequent release of the system adds function to the previous release until all designed functionality has been implemented.



- The system is put into production when the first increment is delivered.
- The first is increment a core product where the basic requirements are addressed.
- Supplementary features are added in the next increments one by one
- Requirements are taken only once and locked them, and remaining steps will be implemented iteratively.

Advantages

- * Module-by-module working
- * Customer interaction maximum
- * used for large projects
(Web based / product based)
- * flexible to change the code at any time

Disadvantages

- * No overlapping of phases
- * Requires a good planning design
- * Rectifying a problem in one unit requires correction in all units.

7. Evolutionary Model: krishna ⑨

→ Evolutionary model is a combination of Iterative and Incremental model of SDLC.