Spiral Model: SDLC (software development life cycle)

Spiral Model: SDLC (software development life cycle)

Spiral Model: SDLC (software development life cycle)

- Spiral model was proposed by BOHEM in 1985.
- Due to the curved shape of this model, it is called a spiral model.
- The spiral model includes both the waterfall model and the prototype model,
- that is, it is a combination of both the waterfall model and the prototype model.
- Spiral model is used for large projects, it is not used in small projects and this model is very expensive.

Spiral Model: SDLC (software development life cycle)

- The spiral model works only on prototyping and is also called iterative model
- By following the rule of spiral model, whatever software is developed,
- it develops rapidly by developing new versions of the software in that software.
- Increases the process of developing a software in a series by using the spiral model.
- In starting this model its prototype can be made only during iterations.
- After that the prototypes are prepared by the engineered system only during iterations.

When use Spiral Model

When use Spiral Model

- When spiral model is used. When any software is large and complicated.
- When there is a need for evolution to continue risk.
- It can be used when the requirement of any user or customer is complicated.
- This model is used the most where the requirement is changing frequently. And there should be enough time to develop the software.
- The best future of waterfall model and prototyping model are used inside this model.

What are the Phases in Sprial Model

Spiral Model: SDLC (software development life cycle)

spiral model phases

- The spiral model has the following 4 phases:-

1:-Planning

2:-Risk analysis

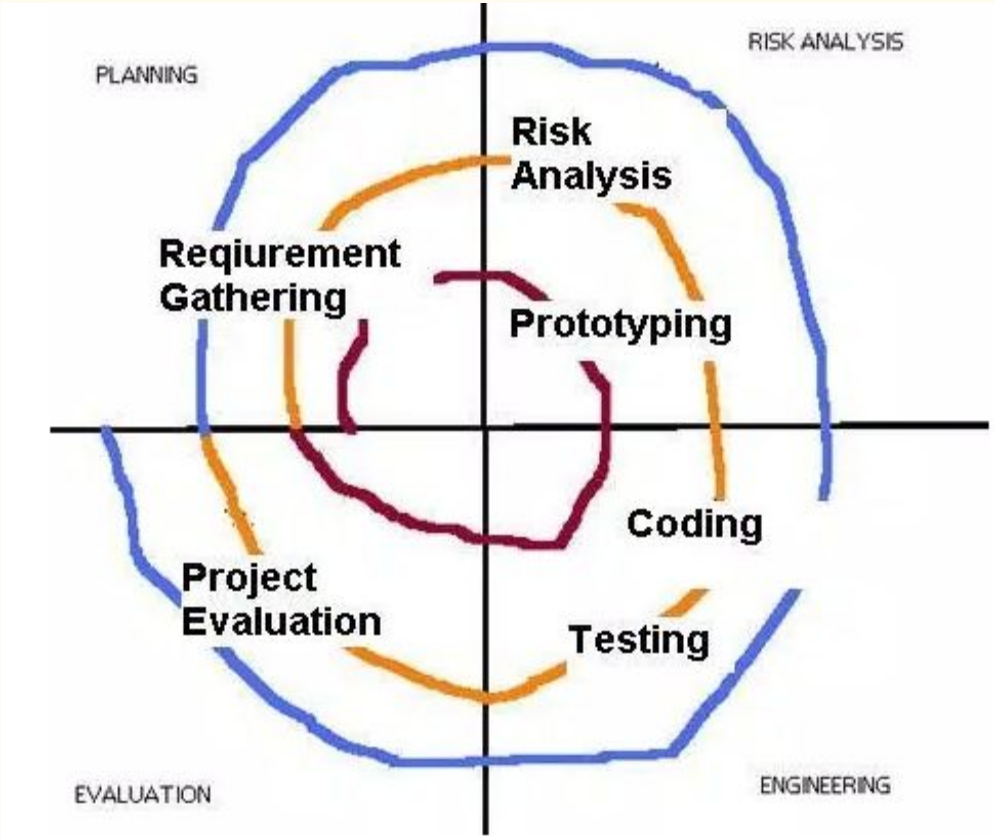
3:-Engineering

4:-Evaluation

Spiral Model: SDLC (software development life cycle)

1:-Planning:-

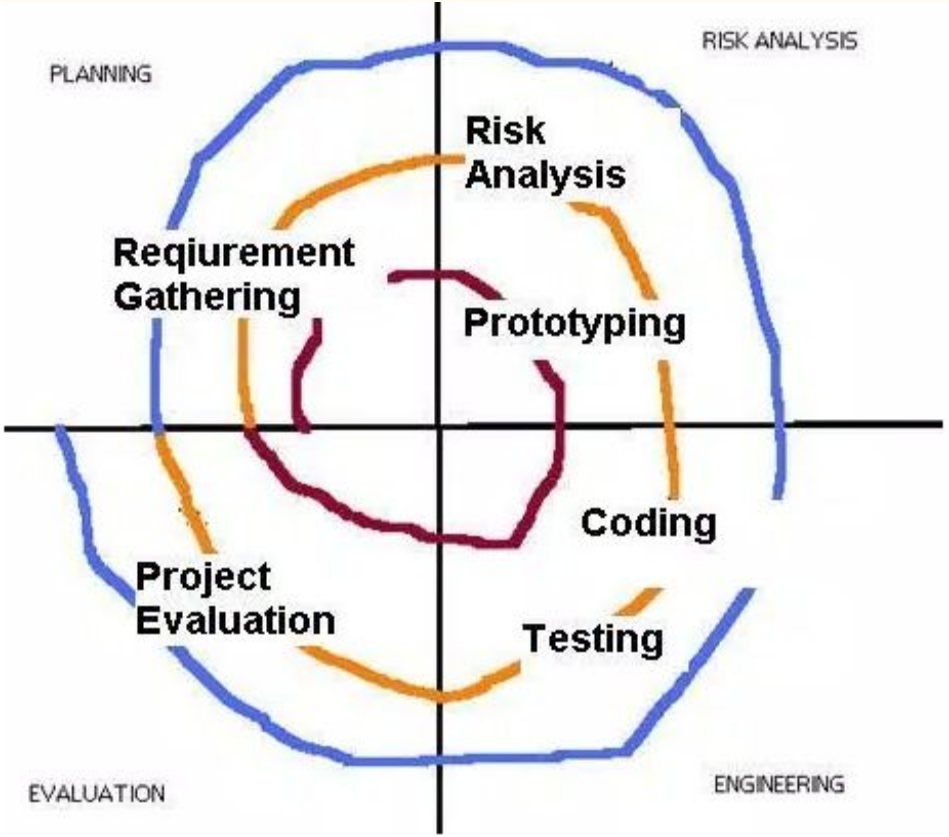
- All our requirements are collected in the planning phase.
- In the planning phase, we discuss what we want the software to achieve or what are its goals?



Spiral Model: SDLC (software development life cycle)

2:-Risk analysis:-

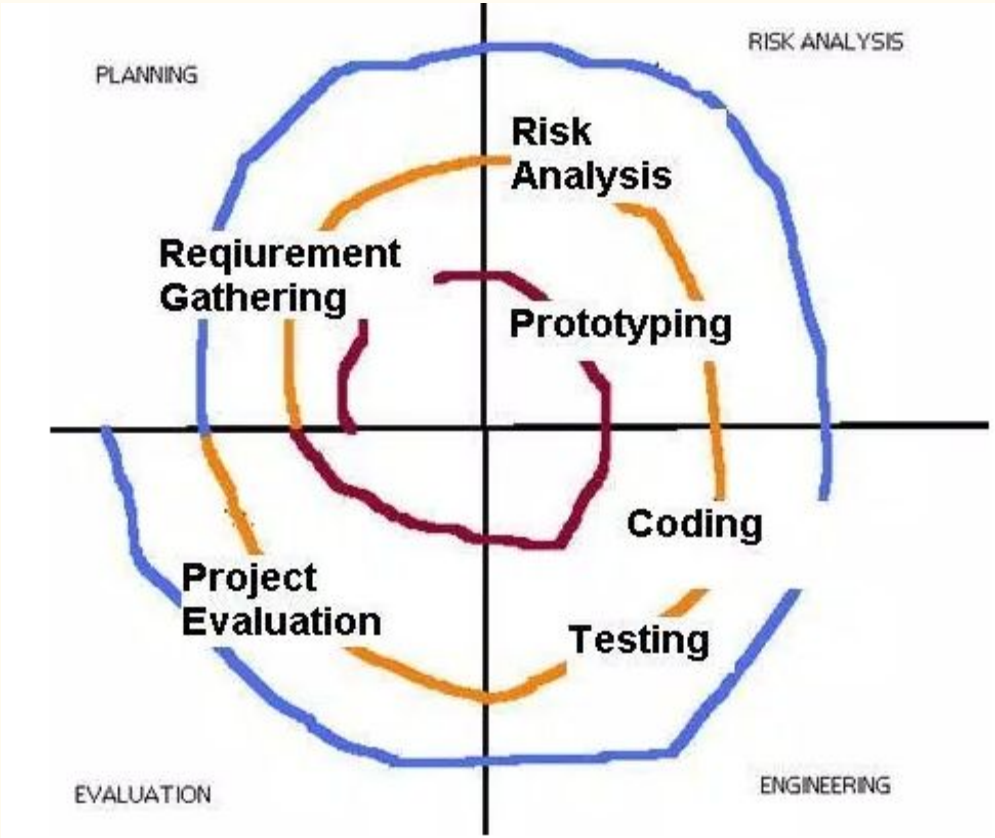
- This phase is a very important phase of the spiral model because risk analysis of any project is done within this phase itself.
- When risk analysis is done in this phase, a prototype is prepared in it.
- If any problem is detected in the risk analysis itself.
- So further iterative of this problem is prepared.



Spiral Model: SDLC (software development life cycle)

3:-Engineering:-

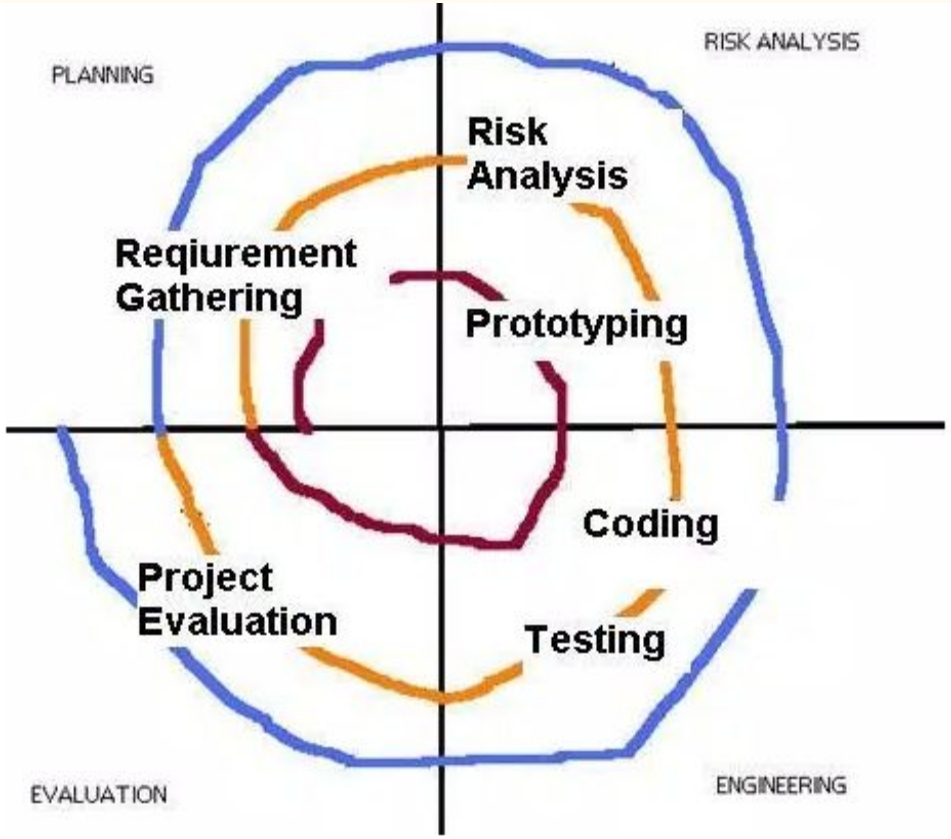
- There is development and testing of software in the engineering phase.
- In which coding and testing are done and software is developed.
- In which the software is developed according to the prototype that has been prepared.



Spiral Model: SDLC (software development life cycle)

4:-Evaluation:-

- In this phase, the customer is evaluated.
- Whether the output that has come is correct or not, there is no need for any update.
- If there is a need for an update, then that update is done there, before going to the next phase.
- Whatever the evaluation phase is, it will be used rapidly till then the correct result will not come.
- Which is the first spiral which happens by analyzing the requirement by locking the requirement in the planning phase in starting.
- This is called the base spiral model.



Sprial Model: Advantages And Disadvantages

Spiral Model: SDLC (software development life cycle)

Advantage of Spiral Model

- This model is the most flexible model because in this the project manager decides whether to keep this project in this development phase, for this the complexity of the requirement has to be understood by manager.
- There is only one phase in this model for doing risk analysis.
- The risks inside the spiral model are very low.
- Best technology has been used inside the spiral model.

Spiral Model: SDLC (software development life cycle)

Disadvantage of Spiral Model

- This model is not used for smaller projects.
- There should be a special expert to analyze the risk in this model.
- The spiral model is very costly because it requires special expert for risk analysis.
- These models are quite complex because risk analysis has to be done again and again.