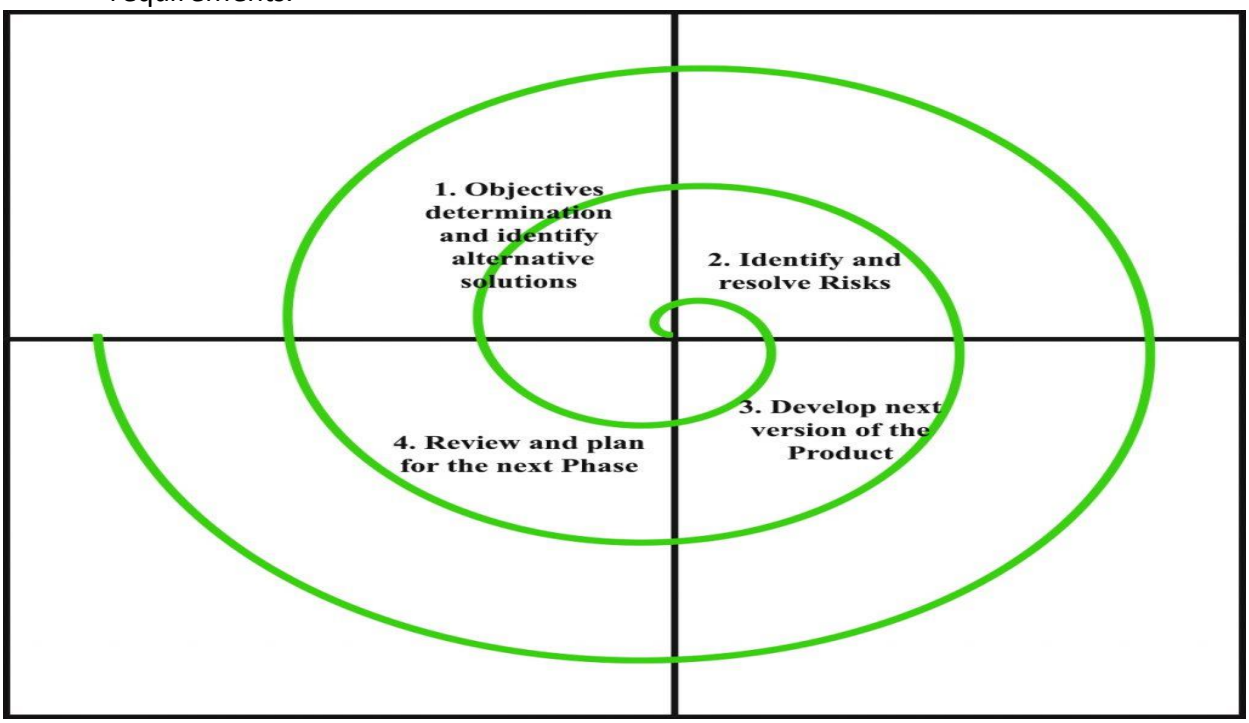
SPIRAL MODEL

- **The Spiral Model** Originally proposed by **Barry Boehm.**
- The spiral model is more emphasized placed on risk analysis .
- Each phase in a spiral model begins with a design goal and ends with the client reviewing the progress.
- The spiral model is 4 phases data remind objectives ,identify and resolve risk , develop and test , plan the next iteration.
- A software project repeatedly passes through these phases in the iterations (call the spirals in this model). The baseline is spiral starting in the planning phase requirements are gathered and risk are assessed.
- the development team in a spiral SDLC model is start with a small set of requirements and goes through each development phase for those set of requirements.



- **Objectives determination and identify alternative solutions:** Requirements are gathered from the customers and the objectives are identified, elaborated and analysed at the start of every phase.Requirements like BRS that is business requirement specification and SRS that is system requirements specification. Then alternative solutions possible for the phase are proposed in this quadrant.
- **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 is identified and the risks are resolved using the best possible strategy. At the end of this quadrant, Prototype is built for the best possible solution.

- **Develop and test 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.
- **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.

Spiral Model – Application

- When the project is large.
- where the software needs continuous risk evaluation.
- where requirements are a bit complicated and require continuous clarification.
- where software requires significant changes.
- where enough timeframe is there to get and user feedback.
- where releases are required to be frequent.

Waterfall Model - Advantages

- Development is fast.
- Larger projects/software are created and handled in a strategic way.
- Risk evaluation is proper.
- Control towards all the phases of development.
- More and more features are added in a systematic way.
- Software is produced early.
- Has room for customer feedback and the changes are implemented faster.

Waterfall Model - Disadvantages

- Disk analysis is important feature requires expert people.
- is not beneficial for smaller projects.
- spiral may go infinitely.
- documentation is more as it has intermediate phases.
- it is costly for smaller projects.