Process Models: Perspective Process Models (Continue)

Lecture # 7

Lecture by Engr.Sidra

Iterative Model

- In the Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.
- An iterative life cycle model does not attempt to start with a full specification of requirements.

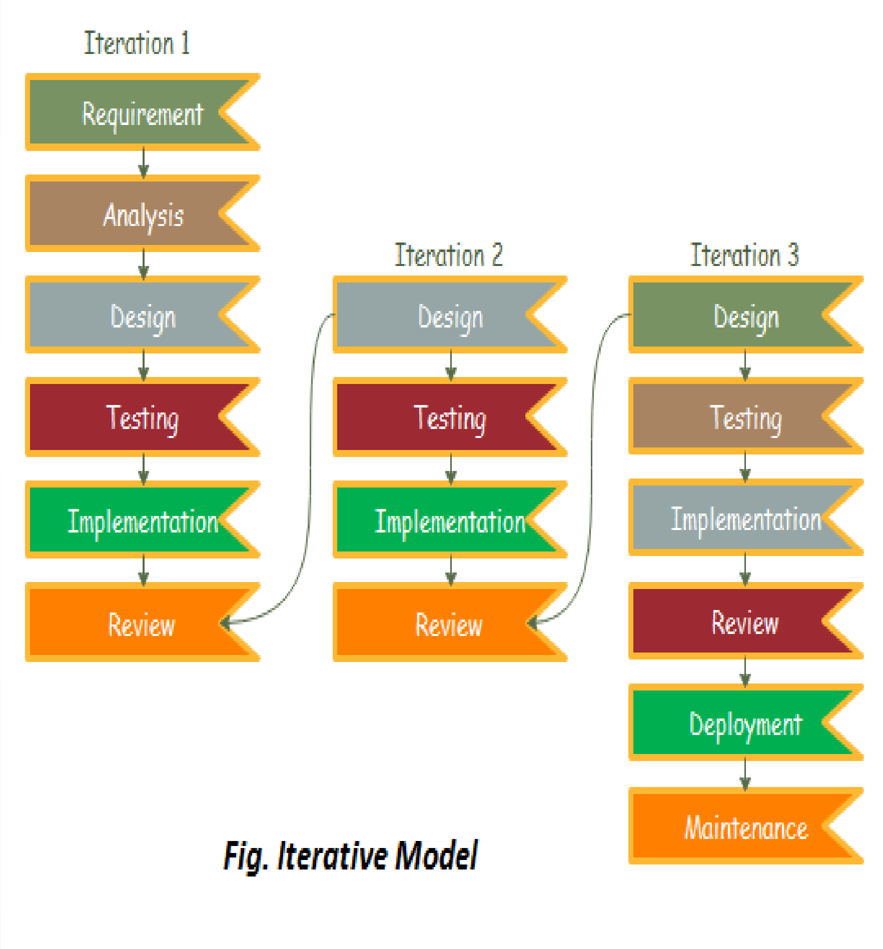


Fig. Iterative Model

Iterative Model

- Instead, development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements.
- This process is then repeated, producing a new version of the software at the end of each iteration of the model.

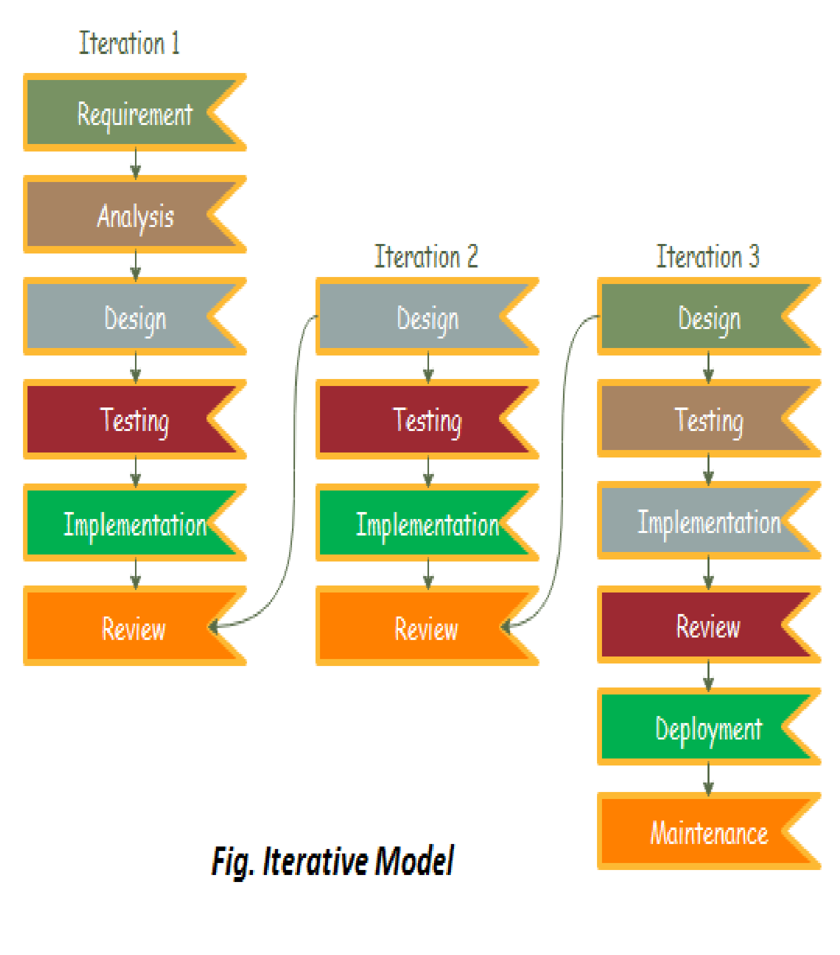
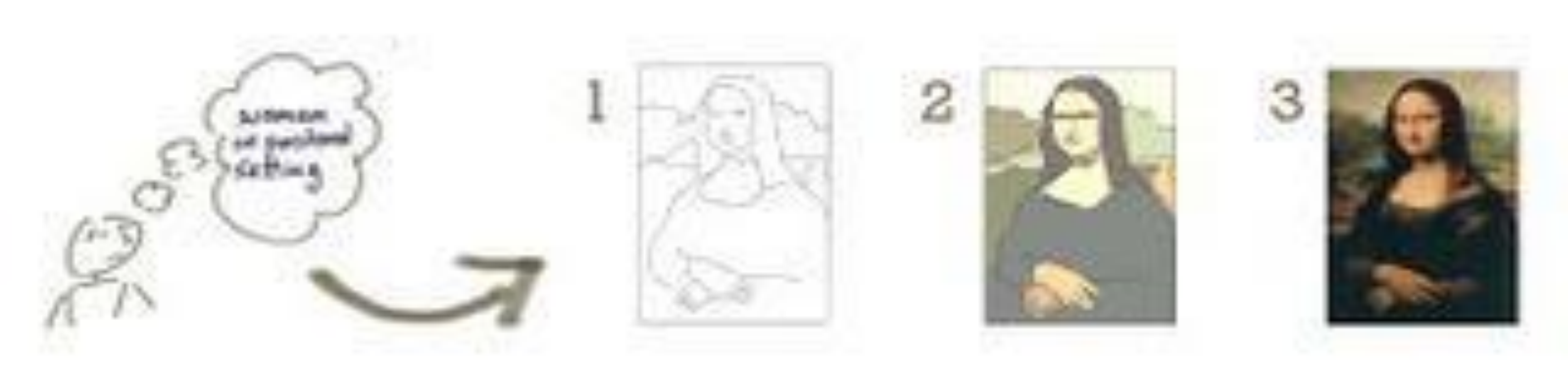


Fig. Iterative Model

Iterative Model - Example

- When we work **iteratively** we create rough product or product piece in one iteration, then review it and improve it in next iteration and so on until it's finished.
- As shown in the image below in the first iteration the whole painting is sketched roughly, then in the second iteration colors are filled and in the third iteration finishing is done.
- Hence, in iterative model the whole product is developed step by step.



Iterative Model - Application

- Major requirements must be defined; however, some functionalities or requested enhancements may evolve with time.
- A new technology is being used and is being learnt by the development team while working on the project.
- Resources with needed skill sets are not available and are planned to be used on contract basis for specific iterations.
- There are some high-risk features and goals which may change in the future.

Advantages of Iterative Model

- Some working functionality can be developed quickly and early in the life cycle.
- Results are obtained early and periodically.
- Parallel development can be planned.
- This avoids the downward flow of the defects.
- Less time is spent on documenting and more time is given for designing.
- Progress can be measured.

Advantages of Iterative Model

- Less costly to change the scope/requirements.
- Testing and debugging during smaller iteration is easy.
- With every increment, operational product is delivered.
- Issues, challenges and risks identified from each increment can be utilized/applied to the next increment.
- Better suited for large and mission-critical projects.
- During the life cycle, software is produced early which facilitates customer evaluation and feedback.

Disadvantages of Iterative Model

- More resources may be required.
- Although cost of change is lesser, but it is not very suitable for changing requirements.
- System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.
- Defining increments may require definition of the complete system.

The image part with relationship ID rId5 was not found in the file.

Disadvantages of Iterative Model

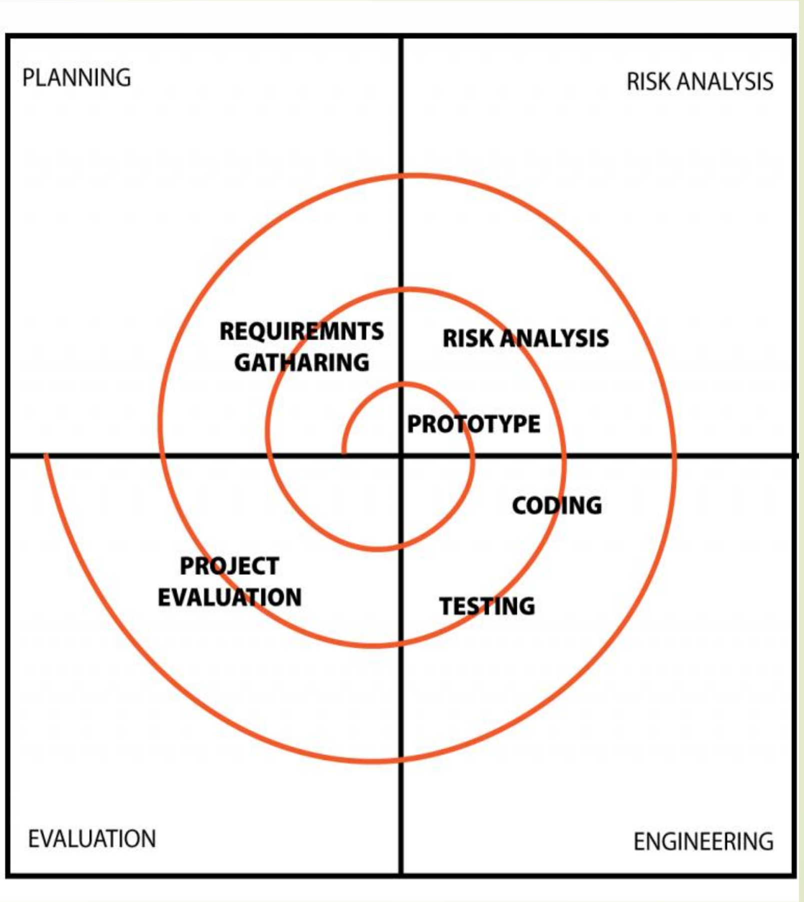
- Not suitable for smaller projects.
- Management complexity is more.
- End of project may not be known which is a risk.
- Highly skilled resources are required for risk analysis.
- Projects progress is highly dependent upon the risk analysis phase.

Spiral Model

- This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model
- It allows incremental releases of the product or incremental refinement through each iteration around the spiral.
- The spiral model is similar to the incremental model, with more emphasis placed on risk analysis.
- The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). The baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed. Each subsequent spirals builds on the baseline spiral.

Spiral Model - Design

- **Planning Phase:**
  - Requirements are gathered during the planning phase. Requirements like ‘BRS’ that is ‘Bussiness Requirement Specifications’ and ‘SRS’ that is ‘System Requirement specifications’.
- **Risk Analysis:**
  - In this phase a process is undertaken to identify risk and alternate solutions.
- **Engineering Phase:**
  - In this phase software is developed, along with testing at the end of the phase.
- **Evaluation Phase:**
  - This phase allows the customer to evaluate the output of the project to date before the project continues to the next spiral.



Spiral Model - Application

- When there is a budget constraint and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
- Customer is not sure of their requirements which is usually the case.
- Requirements are complex and need evaluation to get clarity.
- New product line which should be released in phases to get enough customer feedback.
- Significant changes are expected in the product during the development cycle.
- When costs and risk evaluation is important.

Advantages of Spiral Model

- Changing requirements can be accommodated.
- Allows extensive use of prototypes.
- Requirements can be captured more accurately.
- Users see the system early.
- Development can be divided into smaller parts and the risky parts can be developed earlier which helps in better risk management.

Advantages of Spiral Model

- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Good for large and mission-critical projects.
- Strong approval and documentation control.
- Additional Functionality can be added at a later date.
- Software is produced early in the software life cycle.

Disadvantages of Spiral Model

- Management is more complex.
- End of the project may not be known early.
- Not suitable for small or low risk projects and could be expensive for small projects.
- Process is complex
- Spiral may go on indefinitely.

Disadvantages of Spiral Model

- Large number of intermediate stages requires excessive documentation.
- Can be a costly model to use.
- Risk analysis requires highly specific expertise.
- Project's success is highly dependent on the risk analysis phase.
- Doesn't work well for smaller projects.