PROTOTYPE MODEL:

The Prototype Model that involves creating a prototype of the system or application before building the actual product.

Phases:

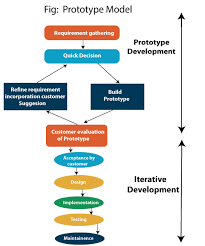
1. Requirements gathering: Identify the basic requirements of the system
2. Quick design: Create a basic design of the system.
3. Build prototype: Build a working prototype of the system.
4. User evaluation: Test the prototype with users to gather feedback.
5. Refine prototype: Refine the prototype based on user feedback.
6. Implement final system: Build the final system based on the refined prototype.

Advantages:

* Improved user satisfaction
* Early detection of errors
* Reduced risk
* Increased flexibility

Dis-advantages:

* Can be time-consuming and costly
* May lead to scope creep
* Requires significant user involvement



Iterative model:

The Iterative Model that emphasizes iterative and incremental development. It involves breaking down the development process into smaller cycles, with each cycle building on the previous one.

Phases:

1. Planning: Define project scope, goals, and deliverables.
2. Analysis: Gather requirements and define system specifications.
3. Design: Create a detailed design of the system.
4. Implementation: Develop and test the system.
5. Testing: Test the system and identify defects.
6. Deployment: Deploy the system to production.
7. Evaluation: Review and assess the system

Advantages:

* Flexibility to changing requirements.
* Reduced risk
* Continuous improvement
* Improved customer satisfaction
* Faster time-to-market

Dis-advantages:

* Requires significant customer involvement
* Can be challenging to manage and co-ordinate
* May lead to scope creep if not managed properly



Spiral Model:

The Spiral Model is that combines elements of both iterative and waterfall models. It is a risk-driven approach that emphasizes incremental releases and continuous improvement.

Phases:

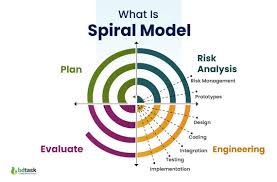
1. Planning: Define goals, objectives, and Deliverables.
2. Risk Analysis: Identify and mitigate potential risks.
3. Engineering: Develop and test the system.
4. Evaluation: Review and assess the system.

Advantages:

* Reduces risk by identifying and mitigating potential issues early
* Allows for continuous improvement and adaptation to changing requirements
* Encourages customer involvement and feedback
* Supports iterative and incremental development

Dis-advantages:

* Requires significant customer involvement
* Can be complex and difficult to manage
* May lead to scope creep if not managed properly



Rapid Application Development Model:

The Rapid Application Development Model that emphasizes rapid development and delivery of a working system. It is an iterative and incremental approach that focuses on quick development and continuous improvement.

Phases:

1. Planning: Define goals, objectives, and Deliverables.
2. User design: Gather requirements and create prototypes.
3. Rapid construction: Develop and test the system.
4. Cutover: Deploy the system to production.
5. Maintenance: Refine and improve the system.

Advantages:

* Fast development and delivery
* Improved user satisfaction
* Reduced risk
* Increased flexibility
* Encourage user involvement

Dis-advantages:

* Requires significant customer involvement
* Can be challenging to manage and co-ordinate
* May lead to scope creep if not managed properly

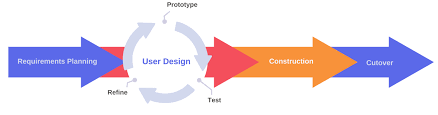


Fig: Rapid Application Development Model