

CONTENTS

Waterfall	2
Advantages	2
Disadvantages	2
V-Model	2
Advantages	3
Disadvantages	3
Incremental	3
Advantages	3
Disadvantages	3
Prototype Model	3
Advantages	4
Disadvantages	4
RAD	4
Advantages	4
Disadvantages	4
Spiral	4
Advantages	5
Disadvantages	5
Agile	5
Advantages	5
Disadvantages	5
Extreme Programming (XP)	5
Advantages	5
Disadvantages	6
Feature Driven Design FDD	6
Advantages	6
Disadvantages	6
Dynamic Systems Development Method (DSDM)	6
Advantages	6
Disadvantages	7
Adaptive SDLC	7
Advantages	7
Disadvantages	7

WATERFALL

Waterfall model was one of the incredibly early process models to be introduced, it is extremely easy to understand.

In this model, each phase must be completed before the new phase can begin. It is mostly used for small projects with certain requirements.

At the end of each phase a review is taken to confirm that project is progressing in the right place.

Stages of Waterfall Model:

1. Project Planning
2. Requirement Documentation
3. Design
4. Implementation
5. Coding
6. Integration and Testing
7. Maintenance

ADVANTAGES

- Easy to use and understand
- One phase is focused and completed at a time.
- Project is team dependent instead of client dependent.
- Every process and results are well documented.

DISADVANTAGES

- No working product is produced until very last stages.
- Not good for long and complex projects.
- Changes that may arise in the completed phase may cause a lot of problems.
- Errors can't be fixed once the phase is complete.
- Documentation takes a lot of time.

V-MODEL

V-Model is also known as Verification and Validation model. V – Model is similar to Waterfall model. It also follows a sequential path of execution, and one phase must be completed before other can begin.

Stages of V – Model:

1. Project Planning
2. High Level Design

3. Low Level Design
4. Implementation and Coding

ADVANTAGES

- Simple and easy to use.
- Disciplined model, one phase is completed at a time.
- Best for small projects where requirements are well understood.
- Each phase has specific review process.

DISADVANTAGES

- Uncertainty and High Risk.
- Poor model for long projects.
- Once the project is completed, it is difficult to go back and change a functionality.
- Not good for projects with changing requirements.

INCREMENTAL

In incremental model whole project is divided into various cycles, and cycles are further divided into easily manageable modules.

Stages of Incremental Model:

1. Requirements
2. Design and Development
3. Testing
4. Implementation

ADVANTAGES

- Easy to debug during small iterations.
- Working software is built during early stages of project.
- Customer can review each build.

DISADVANTAGES

- Total cost is exceedingly high.
- Complete project should be clear beforehand so that it can be broken down into modules.

PROTOTYPE MODEL

Prototype model is a model which displays the functionality of the product, which is being developed, it gives an idea to the customer about what to expect.

Stages of Prototyping:

1. Quick Plan
2. Model
3. Construct

4. Deliver

ADVANTAGES

- Customers are fully involved.
- Errors can be detected before starting the development.
- Early feedback leads to better product development.

DISADVANTAGES

- Can confuse the customer between prototype and actual product.
- Too much time can get wasted if prototypes are rebuilt again and again.

RAD

RAD – Rapid Application Development model, it is quite similar to incremental model. The components are formed in form of mini projects and then assembled into a working product.

Stages of RAD Model:

1. Business
2. Data
3. Process
4. Application
5. Testing and Modeling

ADVANTAGES

- Quick delivery time.
- Components are reusable.
- Efficient customer interaction and feedback.

DISADVANTAGES

- Only systems that can be broken down into modules are used for RAD.
- High skilled developers are required to work on modules parallelly.
- Management is little complex.

SPIRAL

This model is similar to incremental model and waterfall model with high emphasis on risk analysis.

Development Phases:

1. Planning
2. Risk Analysis
3. Design
4. Engineering
5. Evaluation

ADVANTAGES

- Good for large projects.
- Additional functionalities can be added in next cycle.
- Avoidance of risk is greatly enhanced.

DISADVANTAGES

- Can be costly to use.
- Not good for smaller projects.
- Highly dependent on risk analysis phase.

AGILE

This model is combination of iterative and incremental process models. It focuses on process adaptability and is best for time critical projects due to rapid small releases.

Stages of Agile Development Model:

1. Requirements
2. Design
3. Development
4. Testing
5. Development
6. Review

ADVANTAGES

- Satisfied customer due to fast delivery of working product.
- Regular adaption to changing requirements.
- Late changes are possible.
- Less focus on planning.

DISADVANTAGES

- More risk on maintainability.
- High risk of getting off track since there is minimum documentation.
- Transfer of lead to new member can be quite challenging due to lack of documentation.

EXTREME PROGRAMMING (XP)

Extreme programming is not a model, but a methodology, and is considered one of the most effective in software development. It follows "Test-First" development, and it shares agile principles.

ADVANTAGES

- Constant feedback.

- It creates effective software fast due to regular testing at early stages.
- Open communication within the team helps everyone keep up with the process and progress.

DISADVANTAGES

- It focuses more on code than the design, sometimes this leads to failure.
- Little to no documentation.
- Deadlines are tight, so there is a lot of pressure on the team.

FEATURE DRIVEN DESIGN FDD

FFD's most crucial aspect are features. This is a team-development approach and works best with team having a lot of members, so that roles can be assigned to each of them.

Stages of FFD:

1. Develop
2. Build
3. Plan
4. Design

ADVANTAGES

- Time efficient, since it allows multiple teams to work simultaneously.
- Better process tracking.
- Large projects can be built affectively.

DISADVANTAGES

- Emphases on individual code ownership.
- No written documentation to the client.
- Not ideal for smaller projects.

DYNAMIC SYSTEMS DEVELOPMENT METHOD (DSDM)

DSDM is an agile framework for the software creation. It is an iterative, incremental approach, based largely on the technique of Rapid Application Development (RAD).

Stages of DSDM:

1. Feasibility Study
2. Function model
3. Design and build iteration
4. Implementation

ADVANTAGES

- On-time delivery.
- Easy progress tracking.
- Flexible in terms of requirement evolution.

DISADVANTAGES

- Expects continuous user involvement.
- Highly skilled development team required.

ADAPTIVE SDLC

Adaptive SDLC methods provide a combination of incremental and iterative development. It includes incrementally adding functions and making feedback-based improvements and refinements. In other words, the job can easily adjust to the evolving needs based on new customer feedback.

Stages of Adaptive SDLC:

1. Speculate
2. Collaborate
3. Learn

ADVANTAGES

- Short feedback loops lead to quicker and effective adaptation to changes.
- Agile methodologies are effective in nature and enhance teamwork.

DISADVANTAGES

- Extensive customer involvement throughout the SDLC.
- Frequent changes can result in less detailed documentation.