***SDLC models***

* Waterfall model
* Incremental model
* Iterative model
* V – shaped model
* Spiral model
* Agile model

***Waterfall model***

First, oldest traditional. Is a linear sequential model

A diagram of software development

AI-generated content may be incorrect.

***When and where to use waterfall model***

* Requirements clear and fixed
* Application is not complicated
* Short duration projects
* Enough resources
* Mission critical project

***Pros***

* Simple and easy
* Manageable
* Phases completed one at the time
* Scope is fixed
* Clear phases/ goals/ stages
* Visibility on progress
* Well documentation

***Cons***

* No working software in early stages
* Tracking progress within phase is tough
* Underutilization of resources
* Not fit for project where requirements changes
* Poor model for long term projects
* No scope adjustment

***Incremental model***

* Requirements
  + Desgin and development
    - Testing
    - Implementation
  + Desing and development
    - Testing
    - Implementation
  + Design and development
    - Testing
    - Implementation

***When and where incremental model***

1. Requirements are well understood
2. Limited resources
3. Scope changes to some extent

***Pros***

* Save time
* Slight scope adjustment
* Error identified early
* Feedback

***Cons***

* Late customer review
* Rework if scope changes

***Iterative model***

Iterations: refining something repeatedly unit satisfied

***Planning -> (analysis -> design -> development -> testing) -> release***

***When and where iterative model***

1. Startup projects or product based companies
2. When all projects requirements are not clear
3. When we don’t have many resources available

***Pros and cons of iterative model***

***Pros***

* Working software in early stages
* Early feedback
* Flexible scope and requirements
* Assist in early decision making

***Cons***

* Impact of requirement change
* Impact of technology changes
* More management attention needed
* System design issues will come later

***V-Model***

A diagram of a software testing process

AI-generated content may be incorrect.

***When and where V-Model***

* Small and medium size projects
* Medical industry projects
* Compliance & accuracy demanding projects

***Pros and cons of V model***

***Pros***

1. Phases completed one at a time
2. Work well for smaller projects
3. Simple and easy to use
4. Easy to manage

***Cons***

1. Not good for complex projects
2. Poor model for long ongoing projects
3. No scope for requirement change
4. Difficult to change functionality at later stage

***Spiral model***

* Avoid ris/ highlight risk at early stages
* Flexible scope and requirements

A diagram of a risk analysis

AI-generated content may be incorrect.

***When & where spiral model***

High risk projects

Risk & cost evalution is priority

Flexible requirement & scope

IT companies

***Pros & cons of spiral model***

***Pros***

* Risk assessment
* Most useful for large & High-risk projects
* Flexible requirement and scope
* Users sees system early
* Easy cost estimation
* Always space for customer feedback

***Cons***

* Expert people required
* No useful for small projects
* Challenging to predict duration of project
* Bad image to customer
* Risk of not meeting schedule & budget

Agile

…