Iterative Model

feasibility study no

2month

planning, analysis, implementation, and evaluation.

adv

feedback system from forward phase to backword

dadv

no overlapping

no immediate delivery

rigid(no changes)

less customer interaction

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*888888

***Concurrent models***

Concurrent models are those models within which the various activities of software development happen at the same time, for faster development and a better outcome. The concurrent model is also referred to as a parallel working model.

***Rapid Application Development***

The Rapid Application Development (or RAD) model is based on prototyping and iterative model with no (or less) specific planning. In general, RAD approach to software development means putting lesser emphasis on planning tasks and more emphasis on development and coming up with a prototype

V-shaped model

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

V-shaped model

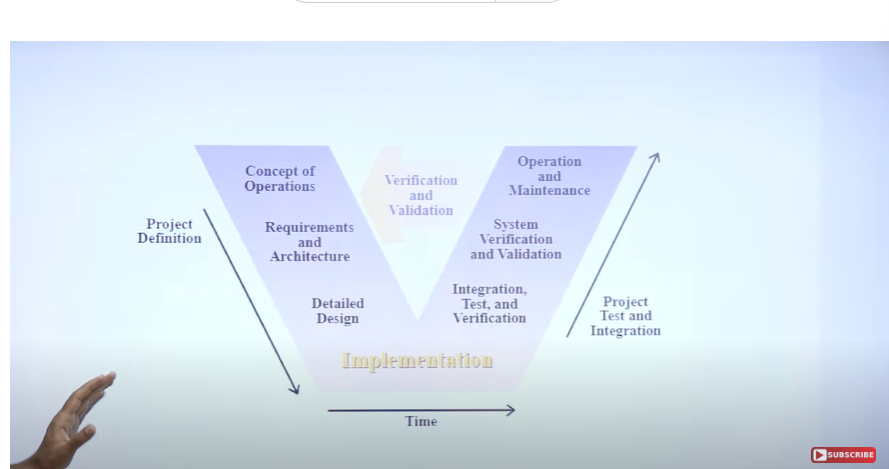
verification and validation model

extension of WF model

testing is associated with every phase of lifecycle

\*)verification phase(requirnment analysis,system design,architecture design,Module design)

\*)validation phase(unit testing,Integration,system,acceptance testing)



Adv:

* Time saving
* Good understanding of project in the beginning.
* Every component must be tested.
* Progress can be tracked easily
* Proactive defect tracking

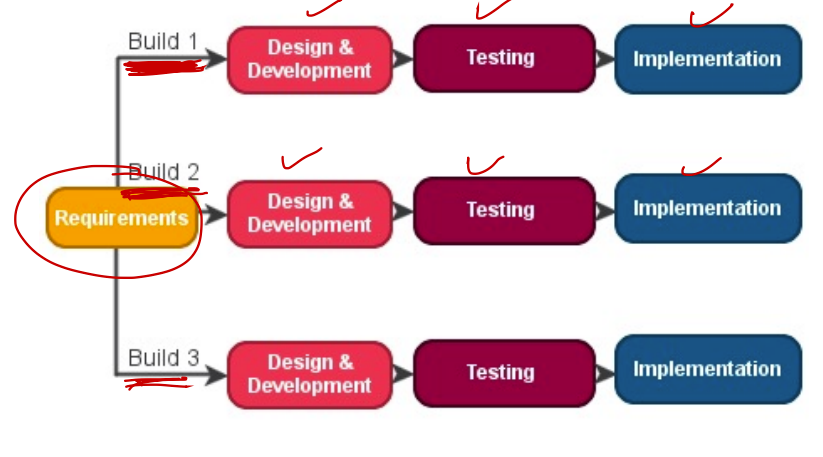
Parallel testing

Diadvantage:

* No feedback so less scope of change
* Risk analysis not done
* Not good for big or object-oriented projects

| COMPARISON FACTORS | FUNCTION ORIENTED DESIGN | OBJECT ORIENTED DESIGN |
| --- | --- | --- |
| **Abstraction** | The basic abstractions, which are given to the user, are real world functions. | The basic abstractions are not the real world functions but are the data abstraction where the real world entities are represented. |
| **Function** | Functions are grouped together by which a higher level function is obtained. | Function are grouped together on the basis of the data they operate since the classes are associated with their methods. |
| **execute** | carried out using  structured analysis and structured design i.e, data flow diagram | Carried out using UML |
| **State information** | In this approach the state information is often represented in a centralized shared memory. | In this approach the state information is not represented is not represented in a centralized memory but is implemented or distributed among the objects of the system. |
| **Approach** | It is a top down approach. | It is a bottom up approach. |
| **Begins basis** | Begins by considering the use case diagrams and the scenarios. | Begins by identifying objects and classes. |
| **Decompose** | In function oriented design we decompose in function/procedure level. | We decompose in class level. |
| **Use** | This approach is mainly used for computation sensitive application. | This approach is mainly used for evolving system which mimics a business or business case. |

***Incremental model***

****

* Customer interactive model
* Large project
* Module by module working
* Early release product demand
* Flexible to change

***Evolutionary Model***

* Combination of iterative and incremental model of sdlc.
* Incremental mode 1st implement a few basic feature and deliver to customer.
* Then build the next part and deliver it again and repeat this step until the desire system is

Fully relized.no long-term plans are made.

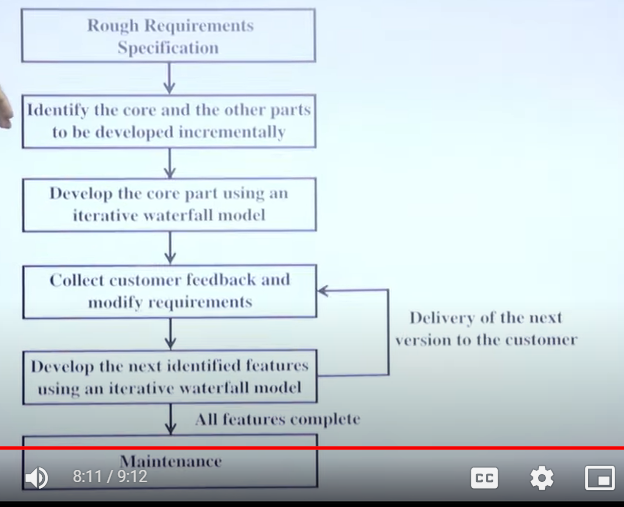
* Also know as Design a little,build a little,test a little,deplioy a little.

**Avd**

* Customer requirements are clearly specified
* Risk analysis better
* It support changing environment
* Initial operation time is less
* Better suited for large mission-critical project

DisAvd

* Not suitable for smaller project
* Cost
* Resources highly skill

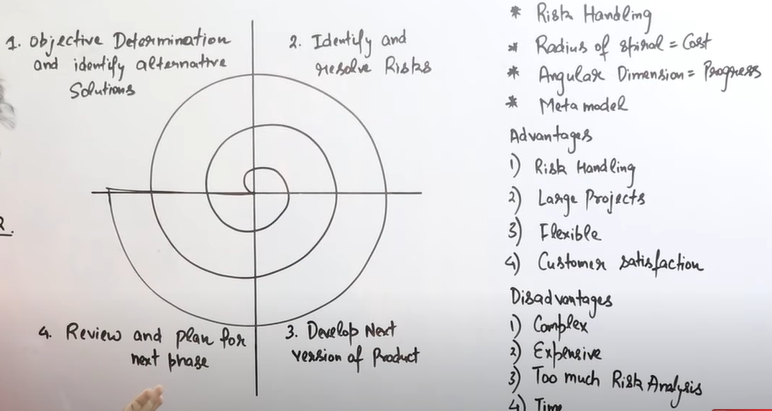


***SPIRAL MODEL***

Focus on risk

Use for large project

Propose by barry boehm



It has angular and radial dimentions

***Agile Methodologies***

Agile Manifesto

§ Individuals and interactions

§ self-organization and motivation are important

§ Working software

§ Demo working software is considered the best means of communication with the customers to understand

their requirements, instead of just depending on documentations

§ Customer collaboration

§ continuous customer interaction is very important to get proper product requirements

§ Responding to change

§ focused on quick responses to change and continuous development

***Scum***

* Scrum isn't a process, it's a framework that facilitates processes amongst other thing
* One of the most popular agile mythology.
* Lightweight, iterative and incremental framework.
* Break down the development phases into stages or cycle called sprints.
* One sprint at a time so maximized time and dedication
* It has scrum master and product owner with constant communication on the daily basis.
* Key word backlog, sprint, daily scrum, scrum master, product Owner

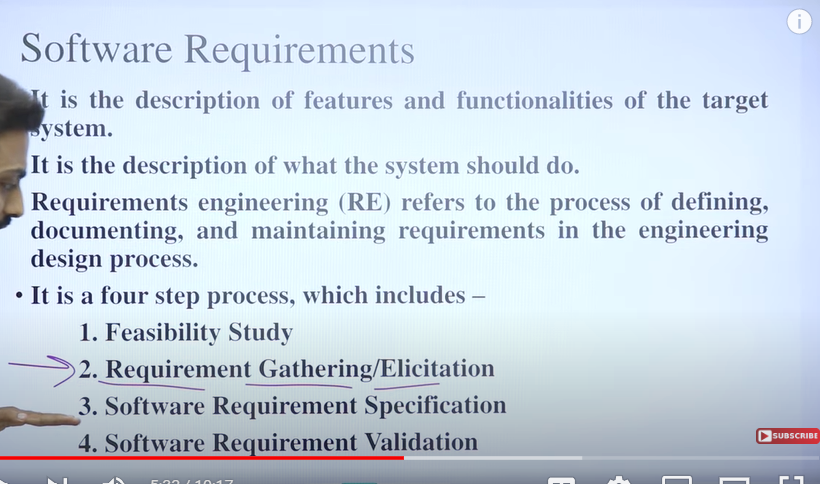
Adv

* reduce development time by40%
* satisfaction is very imp
* freedom and adaption
* revice current sprint and then proceed to next

Disavd

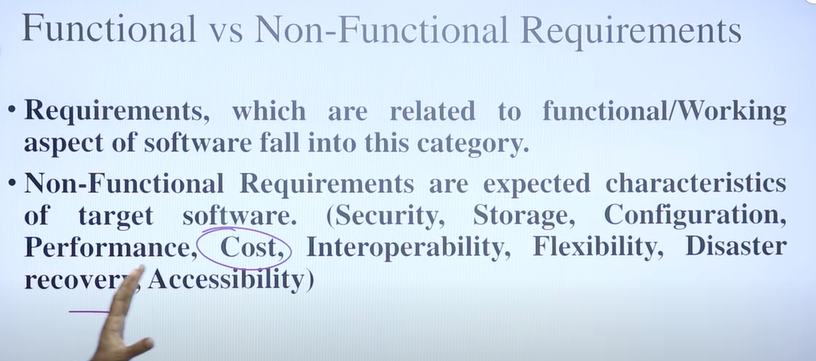
* small team
* no cange in sprint

***COMPARING SDLC MODEL***

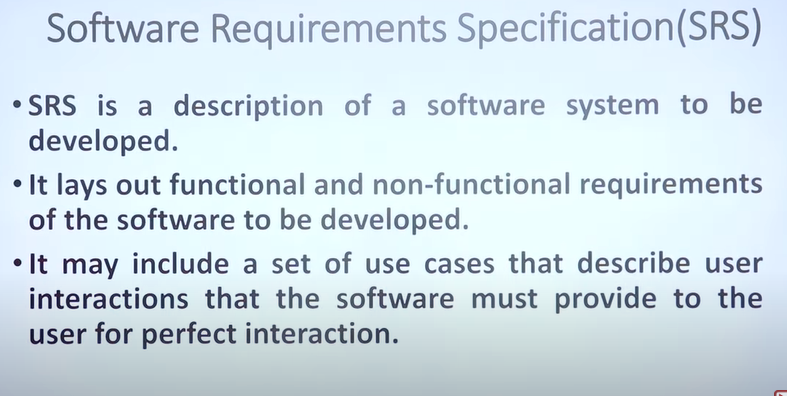




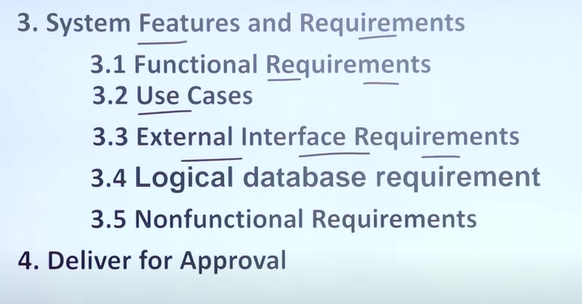
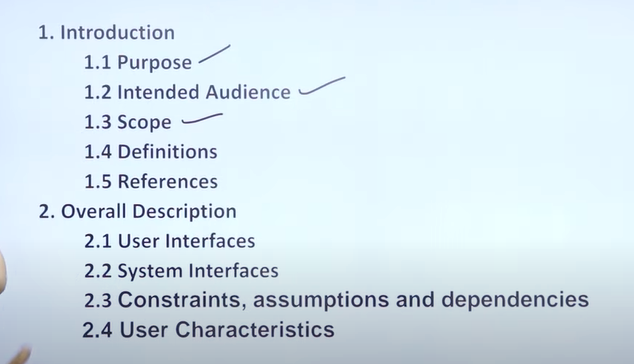
***FUNCTIONAL VS NON-FUNCTIONAL RQUIRNMENT***



***SRS***



***SRS STURUCTURE***

******

