

4<sup>th</sup> Quadrant: - (Plan next phases). (3) (9)

It has one characteristic which is common to all models, the need for advanced technical planning, multiple disciplinary reviews at critical stage or control points.

Advantages: -

1) Estimates: - (ie:- budget, schedule etc) become more realistic as work progress because important issues are discovered earlier.

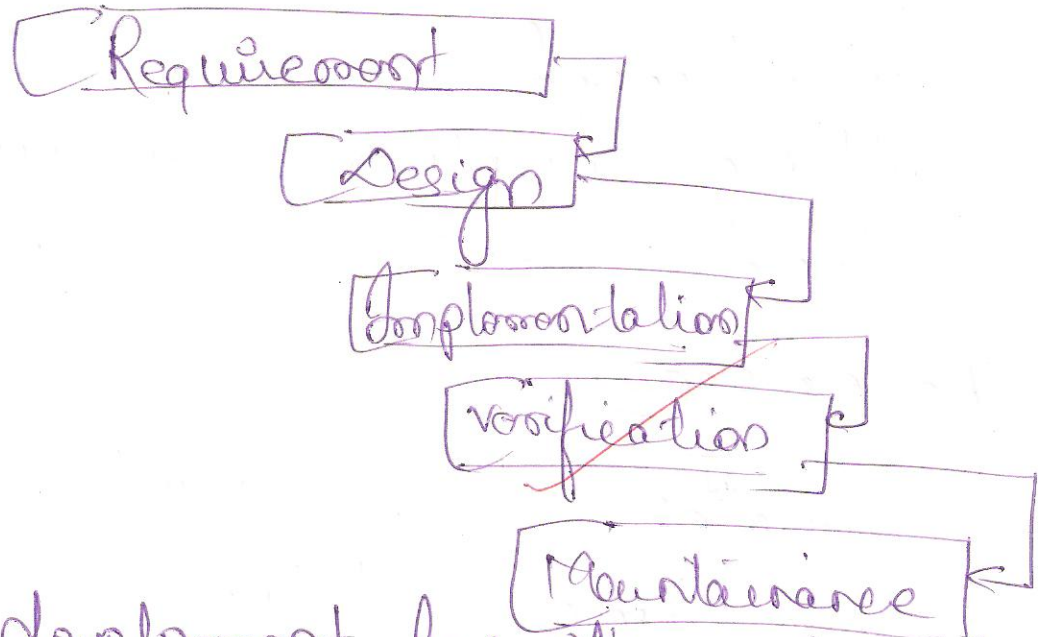
2) It's more able to cope with changes that software development details.

Disadvantages: -

- (1) Highly customized limiting re-usability
- (2) Applied difficulty for each application.
- (3) Risk of not meeting.

(2) Waterfall Model: - It's a sequential software development life cycle process in which progress is seen as following (like a waterfall) through the

④ phases of development, initiation, analysis, design, construction, testing & maintenance.



Waterfall development has its origin in the manufacturing & construction industries, highly structured physical environment in which after the fact changes are prohibitively costly, if not impossible. Since no formal software development methodologies existed at the time, this hardware oriented model was simply adapted for software development.

For Royce's original waterfall model, the following phases are followed in order: -

- 1) Requirement Specification
- (ii) Design
- (iii) Core function
- (iv) Integration
- (v) Testing & Debugging
- (vi) Installation
- (vii) Maintenance

To follow, water fall model one proceeds from phase to the next in a sequential manner. For ex- one first completes requirement specification then proceeds to design & so on.

Advantages: -

- (i) Allows departmentalization & managerial control.

Disadvantages: -

It doesn't allow much reflection or revision once an application is in testing stage, it's very difficult to go back & change something that was not well-thoughtout in concept stage. Alternatives to water fall model include joint application development, rapid development & spiral model.