# **FAST, JAD, RAD**

# **FAST**

Facilitated Application Specification Technique

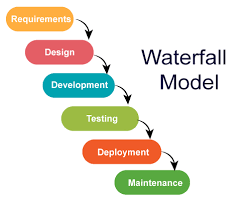
Very clear about the requirement

Customers will not interact with you in between (like 6 months)

## **Waterfall model**

This is in one direction, it cannot go back and redesign as set different requirements

Vincent Roy developed the model in 1970s



### **When does testing start:**

The testing starts from the development phase

Formal testing is done when the

**The type of testing used:**

## **V model**

When a part of the model is completed, it is tested before moving to the next part of the cycle

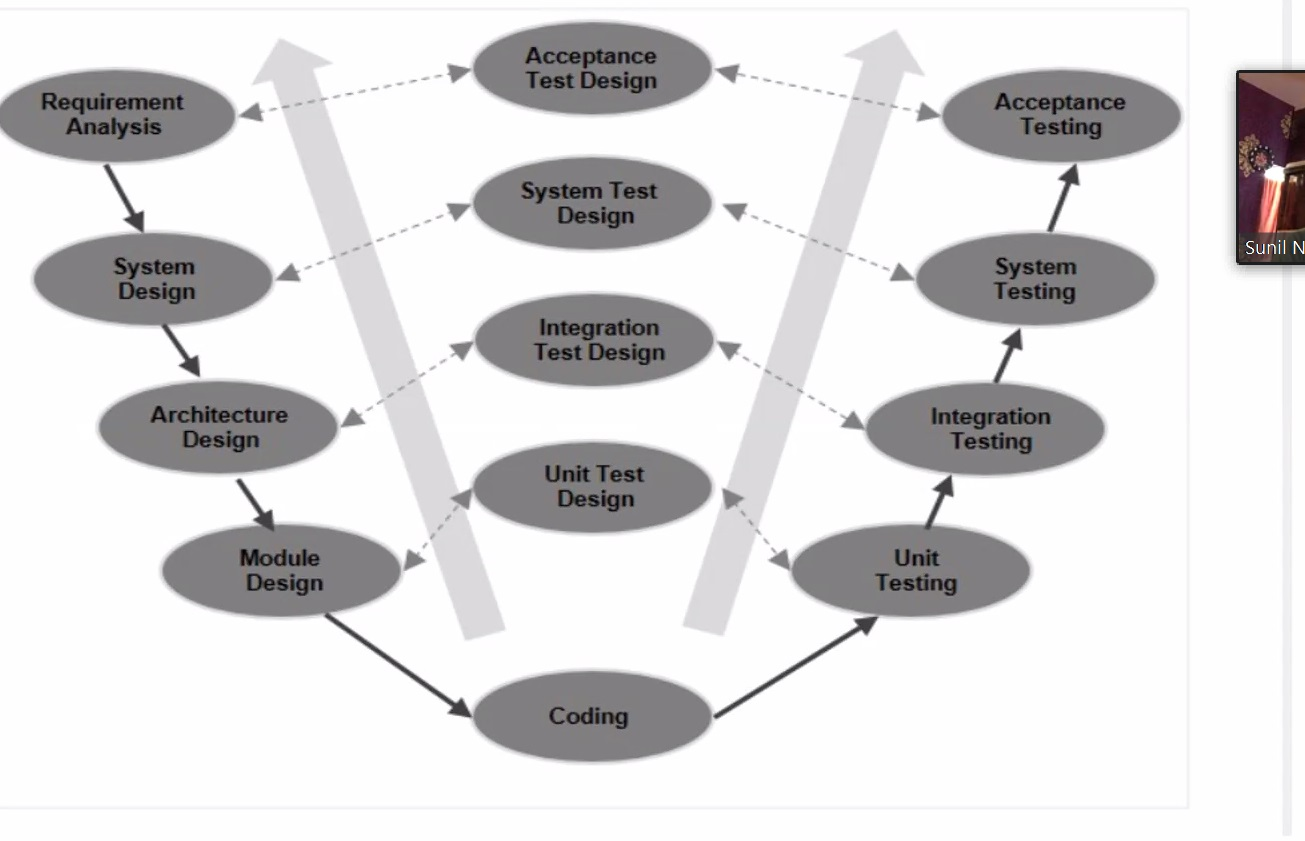
It is aka verification and validation model

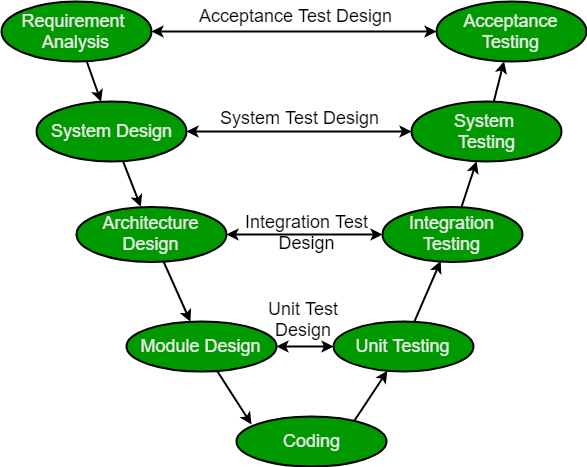
It explains testing very well

the left side of the drawing is aka the development life cycle

the right side is called the testers life cycle

The test case and the requirements are happening simultaneously





### **Merits:**

1. Simple and easy to use easy to implement
2. Several testing activities that are easy to use
3. Proactive defect tracking: defects are found in the early stage that will fix the defects
4. It avoids the downward flow of the defects: it is important to find the defects to avoid the cost increase
5. works well with small projects

### **Demerits:**

1. Hard to execute even if the applications are easy
2. The design has limited flexibility in terms of its execution. It is overall not suitable to use for building object- oriented software
3. It is developed during the phase of implementation, so no initial prototypes of the software are produced
4. Both test documents and requirement documents require to be updated if there is any fault in the system
5. The management of the V model is pretty risky and unstable

## **Difference between Waterfall and V model:**

1. V model: testing starts in requirement phase, can predict the failure earlier
2. If i ask you irrespective of any process model: when does testing start?
3. Make a min

# **JAD**

Joint application development

they are sure of the requirement for that cycle but not for the future cycle, and want to do it in incremental

Customer will be with us through the process

## **Spiral mode:**

It is more about waterfall in a circular motion

Made by Boehm in 1988

This moves in 4 quadrants and do not go back

SDLC: Software Development life cycle makes the spiral model

Spiral models are Step by step process

### **Steps**

1. Feasibility study (Project Manager: decides if we accept the project and the reason why, Business Analyst: Understand the requirement very well and can explain the whole requirement to the development engineer and testing engineer, Architect: the person who will decide which technology we will use, Finance Manager if the project is feasible, HR will assign the number of people to the project)
2. Requirement collection: get the need of the customer which is collected by BA. When we get the requirement, they identify the risk, and mitigate the risk.
3. Design (HLD: it is done while referring to the requirement usually done by the senior development engineer. LLD is the minute details and we do it in reference to the HLD.)
4. Coding: This is done in reference to the LLD
5. Testing
6. Deployment

When we are confirmed of the 1st built requirement and we don’t have clear ideas for the 2nd built requirement

### **When do we go for the Spiral model?**

When there is a dependency between the modules, then we will go with spiral model

Medium to high risk projects

### **Where does this model work best?**

It works well in fin-tech, banking

They will come with the first incremental, and later will bring the next incremental

If you understand the product, and understand its feature, you can work with the product

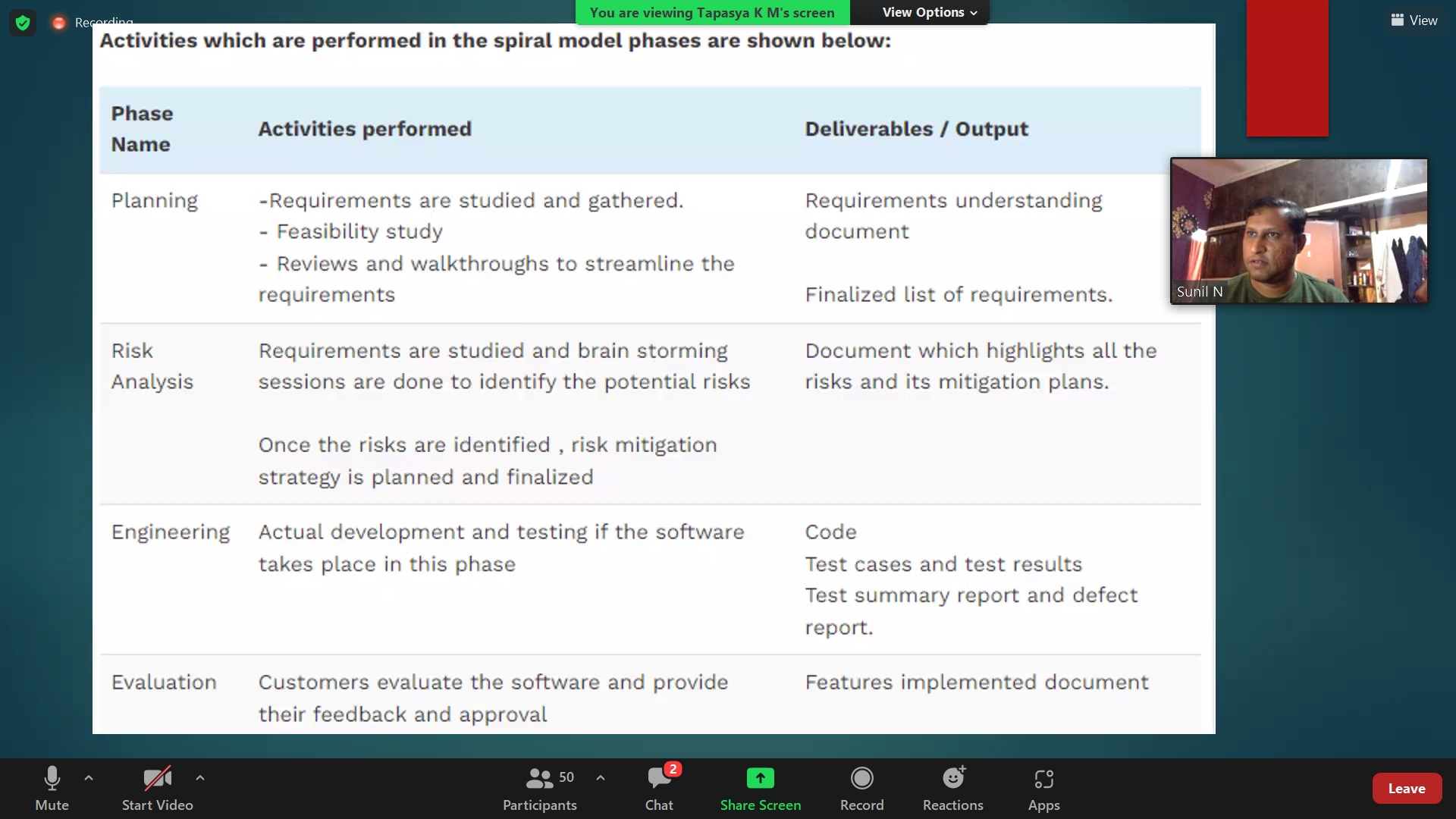
### **Merits:**

1. Flexibility;
2. Risk Management
3. Easy Cost Estimation
4. Good for large and complex projects

### **Demerits**

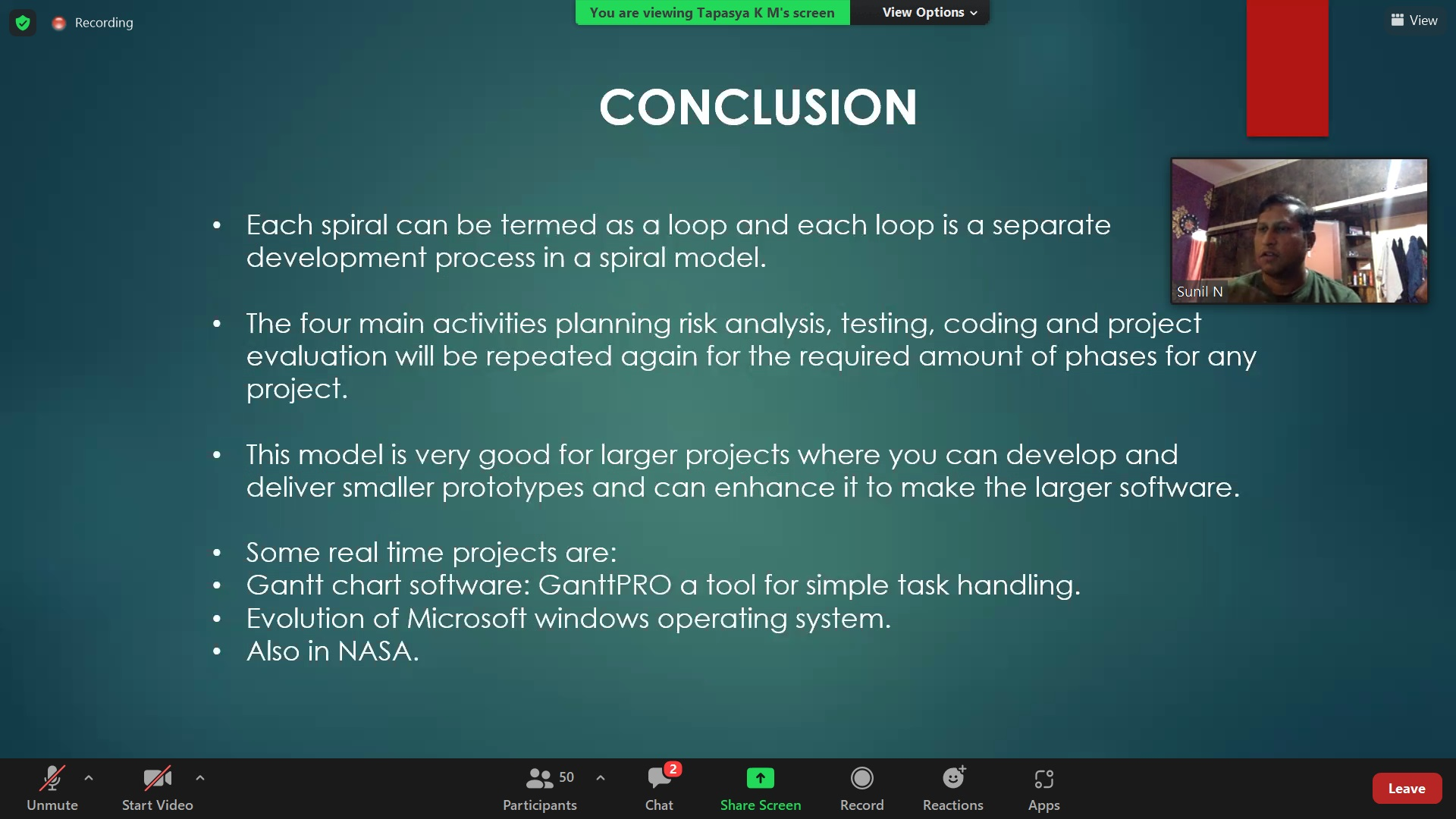
1. Difficult: volume of documentation is large
2. Expensive: Not convenient for small projects
3. Dependability of risk analysis: without highly
4. Difficulty in time management: no. of phases is undiscovered
5. Complicated in terms of user-friendliness

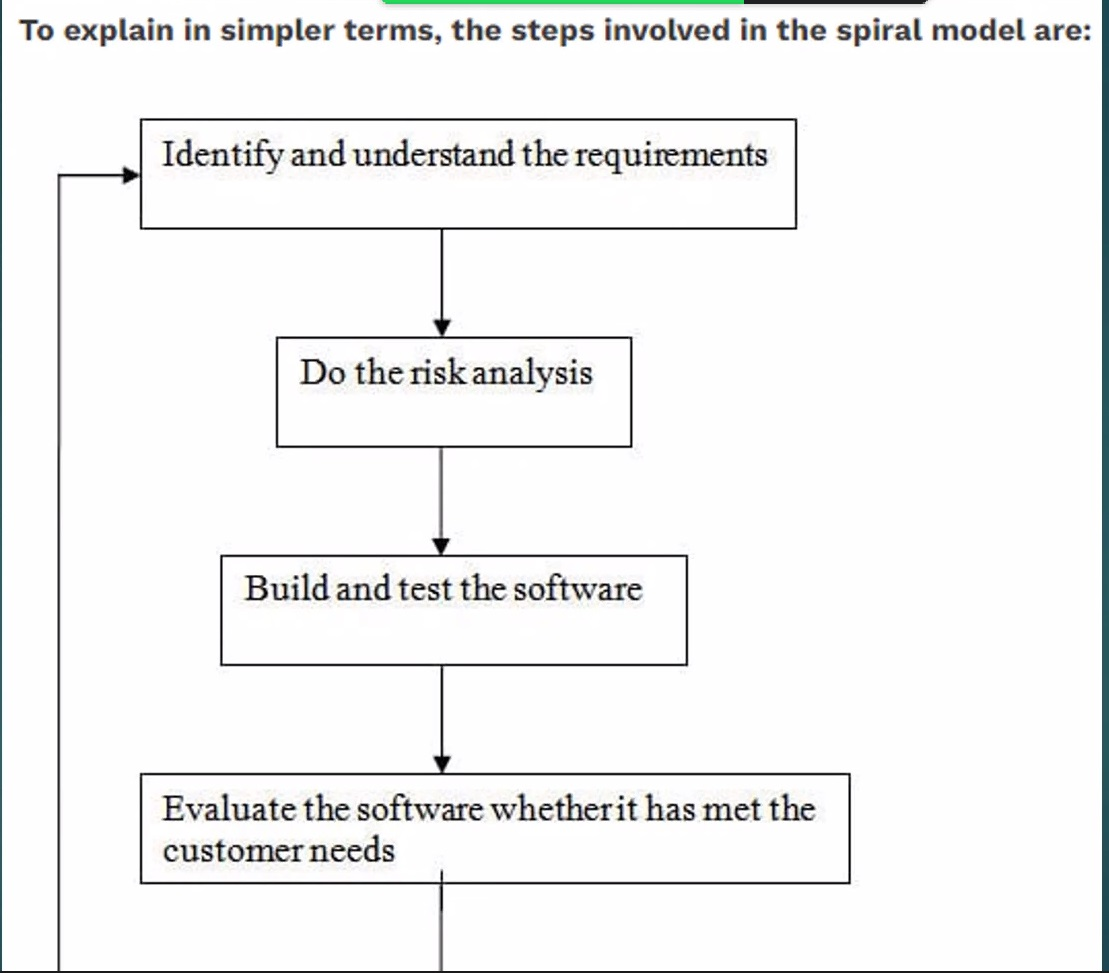
### **Activities**

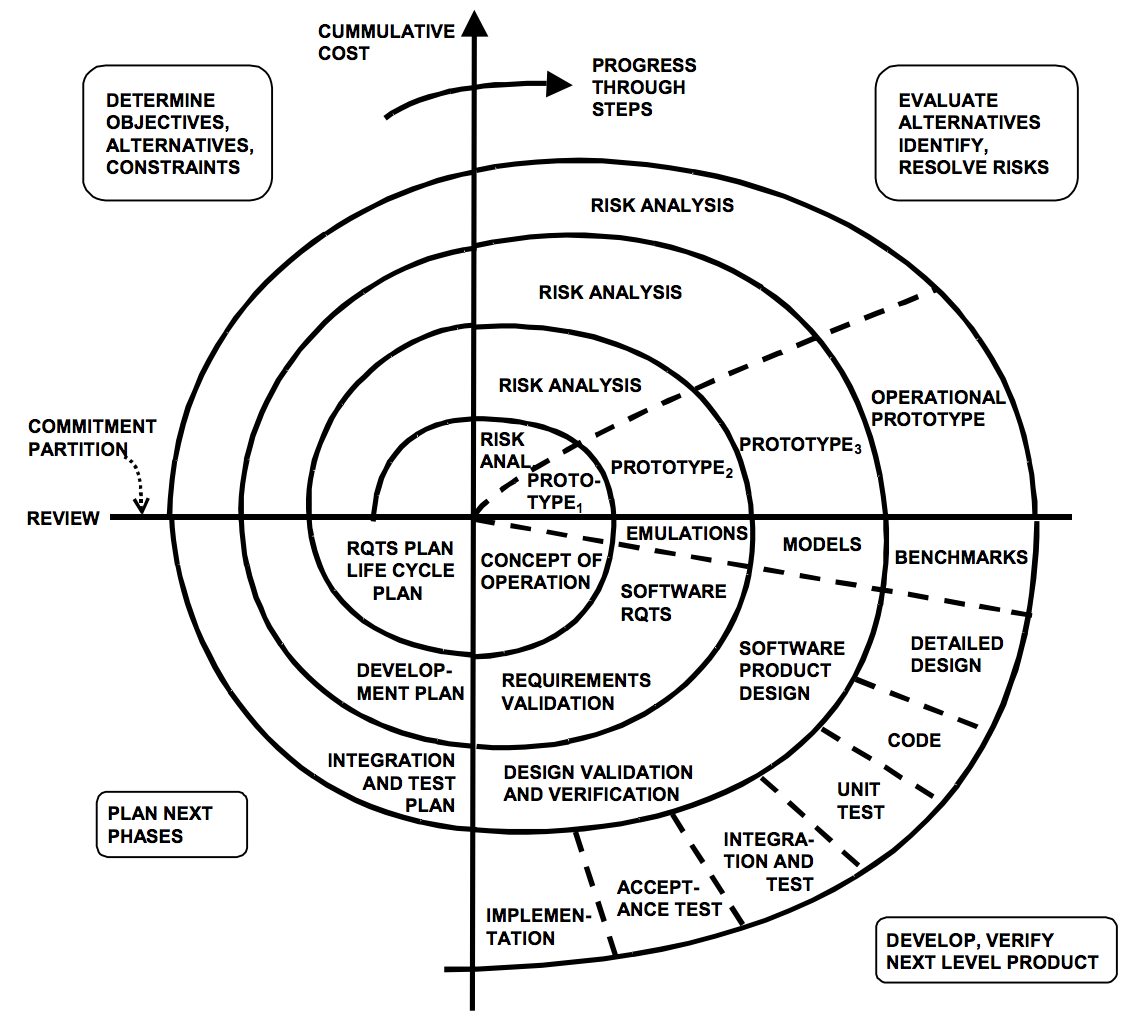
****

### **Conclusion**

1. Each spiral can be termed as a loop
2. each loop is a separate development process is a spiral model
3. 4 main activities: risk analysis, testing , coding, project evaluation







## **Incremental**

It is a type that does one iteration and then a development plan is made for the next iteration.

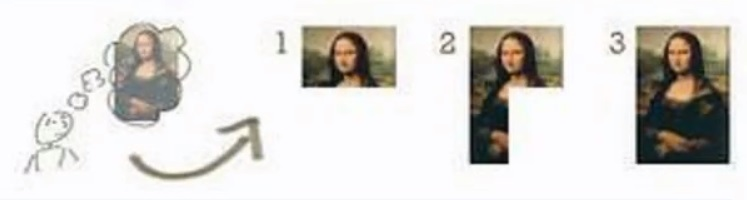
Incremental models are risk free compared to spiral model

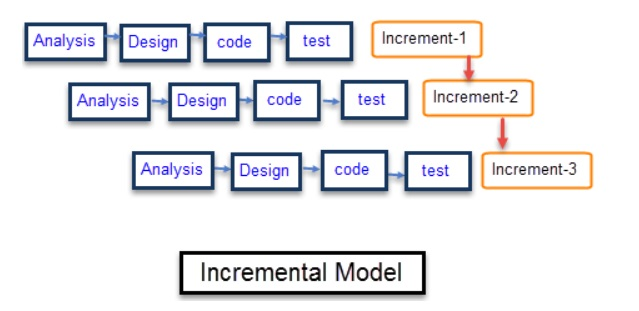
The requirements are given, and they are prioritized

The first build is in 1st 10-20 top priority functions to give to the customers

Then the remaining requirements are met in the following increments

The Incremental model cannot change the requirements till all the builds are released and a new incremental model is made





# **RAD**

Rapid application Development

When the customer is clear about the requirement and needs to see (release) every two weeks and four weeks. The product will be ready in 2 to 4 weeks which is common in the industry and deployed

eg: Whatsapp

Whatsapp used to be only a chat but it increased into status updates. then voice call and worked towards video call etc

Can give you the product though it has defects

## **Agile:**

It is an incremental process but there is no documentation taken

The most used model

It is where the industry is moving

Never come to the conclusion that agile is used in every project

### **Introduction**

Agile: flexibility

Made by 17 in the 2000 to streamline and improve existing models

has incremental aka iterative

Agile calls requirement as Stories

They are confident of the requirement

It speeds up the development process

Agile model one unit is called iteration

which will be delivered in 2-4 weeks

### **Walk Through**

The stages or cycles are called “Sprint” by dividing the development stage

Product owner: responsible person of the whole product, what the product, and the authorized person of what happens to the product

Scrum Master: Will be responsible to the scrum team

Scrum master will plan for a meeting called sprint planning meeting

Sprint planning meeting will have a meeting with the product owner and the scrum team

This meeting will have the number of sprints and what the functionality will be added

Scrum master will ask the team to prepare the sprint backlog

and the backlog is a LLD into sprints and will start working on the product

Daily Scrum: will have the meeting everyday for 5-10 mins

They will make the meeting and have a sprint review meeting with the Scrum master+developer testing

After everyone is satisfied, they will have deployment

And they will have a sprint retrospective meeting with the product owner and all the heads where if they want to change the next sprint

### **Scram technology**

Epics is the HDL (large plan)

Stories are the LLD (steps to get the large plan)

**how agile works:**

1. less documentation as compared to the other models
2. Not so important (documentation)
3. Get the requirement
4. They will have a scrum team
5. Script plan is made by the scrum team
6. Discuss the stories to make up the epic
7. Script planning happens every 2-4 weeks
8. Sprint Backlog is the requirement meeting without the client

### **Merits**

1. After every sprint there is a functionality added
2. If there is change in requirement, the changes are accepted
3. Less Documentation
4. We can have transparency
5. Customer Satisfaction because they are with the project the whole time
6. Late changes are possible with roll backs

### **Demerits**

1. The product may not have a deadline because the product could increase functionality and other processes
2. Difficult for baseline developer because there is no proper documentation and not sure what they are developer
3. small products may not have a lot of functionality
4. Cannot add or leave a team when using agile model, the level of collaboration is not high
5. Less Documentation
6. For complex projects, the resource requirement and effort are difficult to estimate
7. Poor planning of resources

### **Real life projects**

Widely used process

1. Dutch railways wanted to make an app where there was realtime tracking of their passengers

### **2/ 4 weeks cycles for iterations**

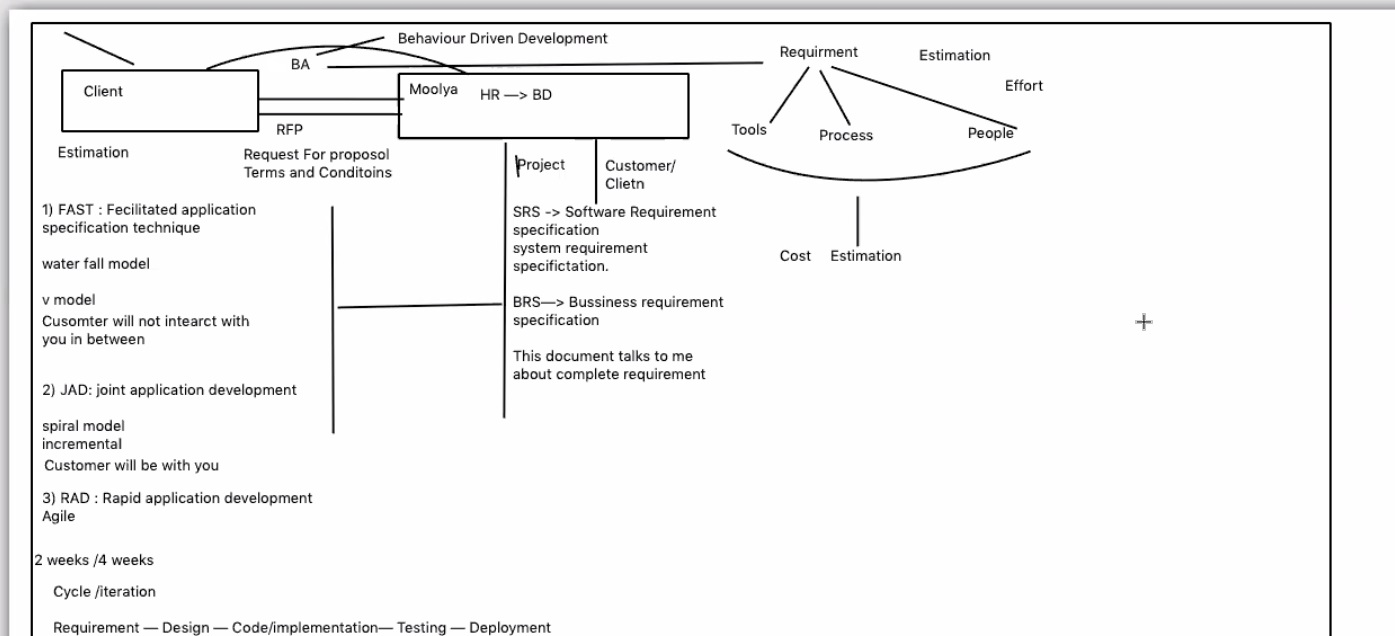
cycle / iteration:

requirement → Design → Code/ implementation → testing → deployment

## **Why is it not easy to change the project framework (FAST, JAD, RAD)**

It is not easy to change the project because some have a working hours of 6 months to 2 weeks, it is hard for the employee

# **Visualization of the definitions**

****

* The Client and Moolya would approach each other somehow
* Then the BA would take in the requirement of the company which would be BDD
* The company would also make a brief requirement for the work
* RFP is done to see if the Client would accept the estimation given by Moolya
* Project begins, where the SRS and BRS is taken (the document that contains the full requirement from the company)
* In the project stage, we will see if we want to go in either of the cycles
  1. FAST
  2. JAD
  3. RAD