

## HOW TO SELECT A SOFTWARE DEVELOPMENT METHODOLOGY

Selecting software development methodology is very important and challenging too. In any software development project there are few things which must be taken care and they are :-

- Risk
- Time to market
- Budget
- Stability of the requirements

One can deal with these things only by selecting a best suited life cycle model for development.

Projects fail because people are unable to manage risk and they are unable to manage risk because methodology they choose is not suitable to manage risks in a project. Other factors also play their role in a failing project if not taken care.

Following questions should be asked for selecting a life cycle model:-

### **Requirements related:-**

Is system proposed a complex system?

Is early/partial delivery functionality a requirement?

Requirements are known or not?

Can the requirements be easily defined?

Can the requirements be defined early in the cycle?

Is there a possibility of change in requirements during project development and is it more likely?

To define requirements will there be a need to do prototype or any other demonstration?

### **Resources related:-**

Is team new to the domain related to project?

Is team new to the technology which is going to be used for project development?

Is team new to the tools/packages to be used for developing the project?

Will team members be assigned to other tasks (other project) during project development?

Availability of training for the team on tools, technologies going to be used in the project?

Ease of resource allocation?

Does the team accept reviews and inspections?

**Customer & end users related:-**

Are the users new to the system?

Will the user's participation required or asked for in the life cycle?

Will the users be trained with the problem domain?

Are users going to participate in all phases of the life cycle?

Is the customer or its representative going to track progress of the project?

**Project & risks related:-**

Is the project an enhancement to an old system?

Are the funds allocated and going to be stable (allocation required) for the project throughout the life cycle?

Is the schedule very strict?

Are module interfaces standard?

Is there a possibility of using reusable components?

Are resources (time, money, tools, people) limited?

Is high reliability required?

Is the modifiability a requirement?

Is it a new product (concept) for the organization?

Is it a system integration project?

You can give weights to these questions (which are actually factors governing selection of a life cycle model) and can give scores for various models based on answer of a question. On the basis of answers various models will get some scores and highest scoring model can be considered.