

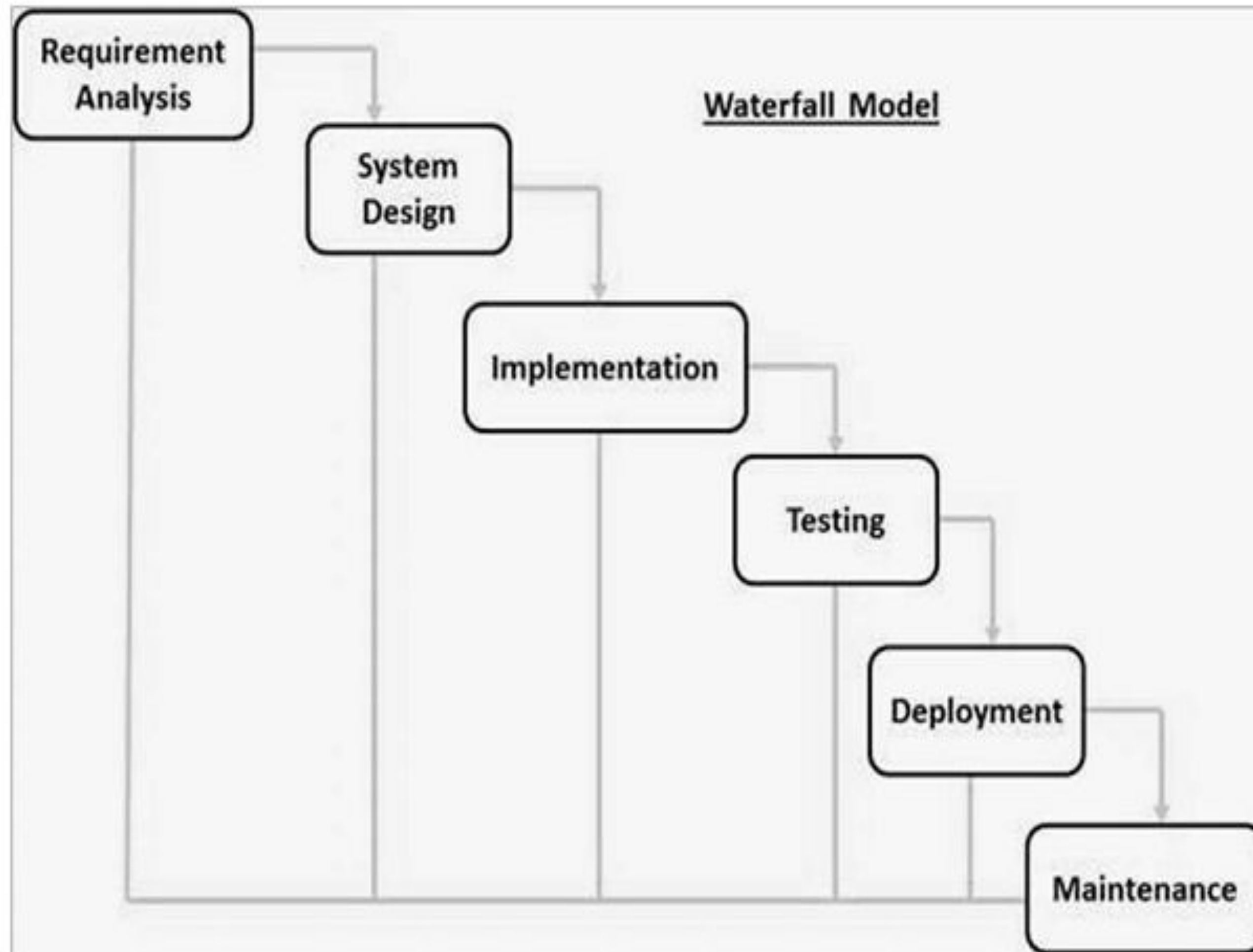
Fundamentals of Software Engineering

Lecture 2 SDLC Models (Process Models)

Waterfall Model

- ❑ The Waterfall Model was the first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**.
- ❑ It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.
- ❑ The Waterfall model is the earliest SDLC approach that was used for software development.
- ❑ The waterfall Model illustrates the software development process in a linear sequential flow.
- ❑ This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

Waterfall Model Cont....



When to use,

- ☐ Project that has clear requirements.
- ☐ Short Projects.
- ☐ Environment is stable
- ☐ Technology and tools used are not dynamic and is stable.
- ☐ Resources are available and trained

Waterfall Model Cont.....

- ❑ **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
- ❑ **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- ❑ **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- ❑ **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- ❑ **Deployment of system** – Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- ❑ **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

Waterfall Model Cont....

Advantages

- ☐ Simple and easy to understand and use
- ☐ Easy to manage due to the rigidity of the model.
- ☐ Each phase has specific deliverables and a review process.
- ☐ Phases are processed and completed one at a time.
- ☐ Works well for smaller projects where requirements are very well understood.
- ☐ Clearly defined stages.
- ☐ Well understood milestones.
- ☐ Easy to arrange tasks.
- ☐ Process and results are well documented.

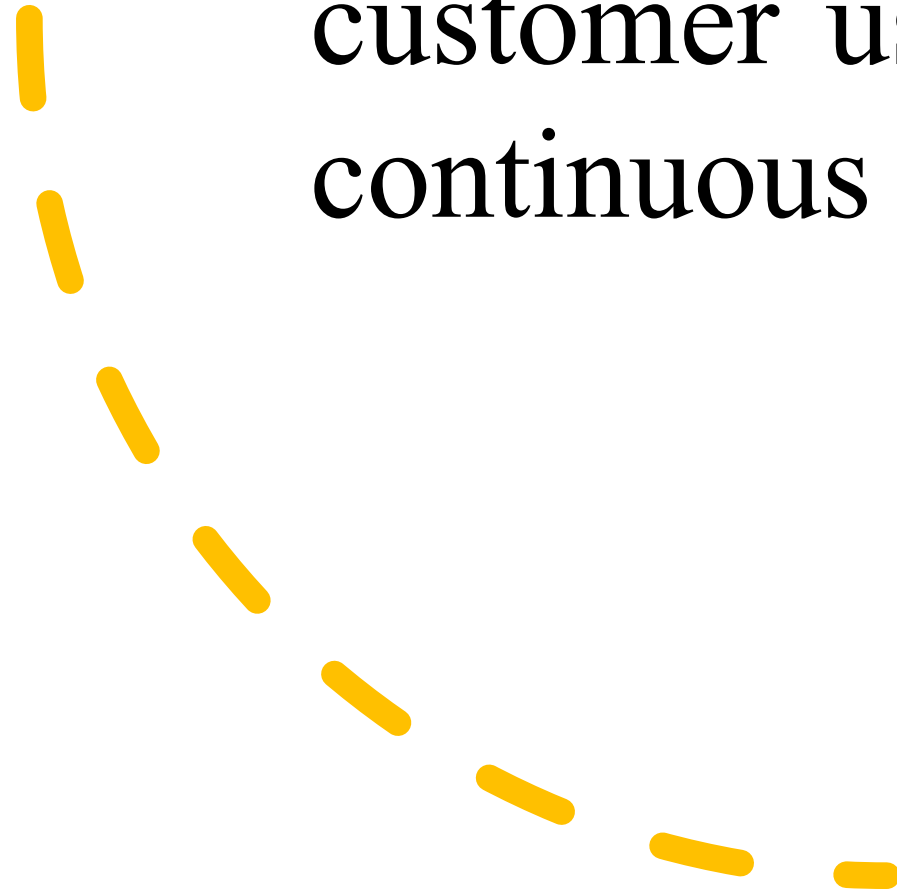
Waterfall Model Cont....

Disadvantages

- ☐ No working software is produced until late during the life cycle.
- ☐ High amounts of risk and uncertainty.
- ☐ Not a good model for complex and object-oriented projects.
- ☐ Poor model for long and ongoing projects.
- ☐ Not suitable for the projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty is high with this process model.
- ☐ It is difficult to measure progress within stages.
- ☐ Cannot accommodate changing requirements.
- ☐ Adjusting scope during the life cycle can end a project.
- ☐ Integration is done as a "big-bang" at the very end, which doesn't allow identifying any technological or business bottleneck or challenges early.



Rapid Application Development Model

- ❑ The **RAD (Rapid Application Development)** model is based on prototyping and iterative development with no specific planning involved. The process of writing the software itself involves the planning required for developing the product.
 - ❑ Rapid Application Development focuses on gathering customer requirements through workshops or focus groups, early testing of the prototypes by the customer using iterative concept, reuse of the existing prototypes (components), continuous integration and rapid delivery.
- 

Rapid Application Development Model Cont....



When to use,

- ☐ When a system needs to be produced in a short span of time (2-3 months).
- ☐ When the requirements are known.
- ☐ When the user will be involved all through the life cycle.
- ☐ When technical risk is less.
- ☐ When there is a necessity to create a system that can be modularized in 2-3 months of time.
- ☐ When a budget is high enough to afford designers for modelling along with the cost of automated tools for code generation.

Rapid Application Development Model

Step 1. Define and finalize project requirements

During this step, stakeholders sit together to define and finalize project requirements such as project goals, expectations, timelines, and budget. When you have clearly defined and scoped out each aspect of the project's requirements, you can seek management approvals.

Step 2: Begin building prototypes

As soon as you finish scoping the project, you can begin development. Designers and developers will work closely with clients to create and improve upon working prototypes until the final product is ready.

Step 3: Gather user feedback

In this step, prototypes and beta systems are converted into working models. Developers then gather feedback from users to tweak and improve prototypes and create the best possible product.

Step 4: Test, test, test

This step requires you to test your software product and ensure that all its moving parts work together as per client expectations. Continue incorporating client feedback as the code is tested and retested for its smooth functioning.

Step 5: Present your system

This is the final step before the finished product goes to launch. It involves data conversion and user training.

Rapid Application Development Model

Advantages

- ☐ Reduced development time.
- ☐ Increases reusability of components
- ☐ Quick initial reviews occur
- ☐ Encourages customer feedback
- ☐ Integration from very beginning solves a lot of integration issues.

Rapid Application Development Model

Disadvantages

- ☐ Depends on strong team and individual performances for identifying business requirements.
- ☐ Only system that can be modularized can be built using RAD
- ☐ Requires highly skilled developers/designers.
- ☐ High dependency on modelling skills
- ☐ Inapplicable to cheaper projects as cost of modelling and automated code generation is very high.

Recap from the lesson learnt

Mark the following True/False

1. RAD model was very popular in earlier but nowadays it is not used.(True/False)
2. The speciality of the waterfall model is one phase can be started after completion of the previous phase. (True/False)
3. Actually the waterfall model is suited when requirements change frequently. (True/False)
4. Waterfall model suits for large projects. (True/False)
5. RAD model is based on prototyping without any specific planning. (True/False)
6. RAD model, a system needs to be produced in a long span of time. (True/False)
7. RAD model is suited When technical risk is less.(True/False)

Spiral Model

- ❑ **Spiral model** is one of the most important Software Development Life Cycle models, which provides support for **Risk Handling**.
- ❑ In its diagrammatic representation, it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project.
- ❑ **Each loop of the spiral is called a Phase of the software development process.** The exact number of phases needed to develop the product can be varied by the project manager depending upon the project risks.
- ❑ As the project manager dynamically determines the number of phases, so the project manager has an important role to develop a product using spiral model.
- ❑ The Radius of the spiral at any point represents the expenses(cost) of the project so far, and the angular dimension represents the progress made so far in the current phase.

Spiral Model Cont.....

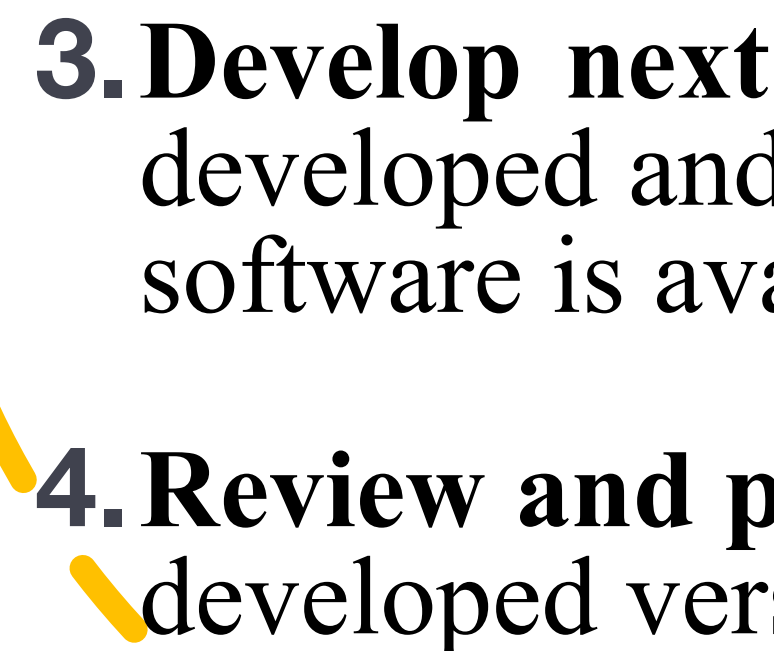


Spiral model is used in the following scenarios:

1. When the project is large.
2. Where the software needs continuous risk evaluation.
3. Requirements are a bit complicated and require continuous clarification.
4. Software requires significant changes.
5. Where enough time frame is there to get end user feedback.
6. Where releases are required to be frequent.



Spiral Model Cont....

- 1.Objectives determination and identify alternative solutions:** Requirements are gathered from the customers and the objectives are identified, elaborated and analysed at the start of every phase. Then alternative solutions possible for the phase are proposed in this quadrant.
 - 2.Identify and resolve Risks:** During the second quadrant all the possible solutions are evaluated to select the best possible solution. Then the risks associated with that solution is identified and the risks are resolved using the best possible strategy. At the end of this quadrant, Prototype is built for the best possible solution.
 - 3.Develop next version of the Product:** During the third quadrant, the identified features are developed and verified through testing. At the end of the third quadrant, the next version of the software is available.
 - 4.Review and plan for the next Phase:** In the fourth quadrant, the Customers evaluate the so far developed version of the software. In the end, planning for the next phase is started.
- 

Spiral Model Cont.....

Advantages

- ☐ Risk Handling
- ☐ Good for large projects
- ☐ Flexibility in Requirements
- ☐ Customer Satisfaction

Disadvantages

- ☐ Complex
- ☐ Expensive
- ☐ Too much dependable on Risk

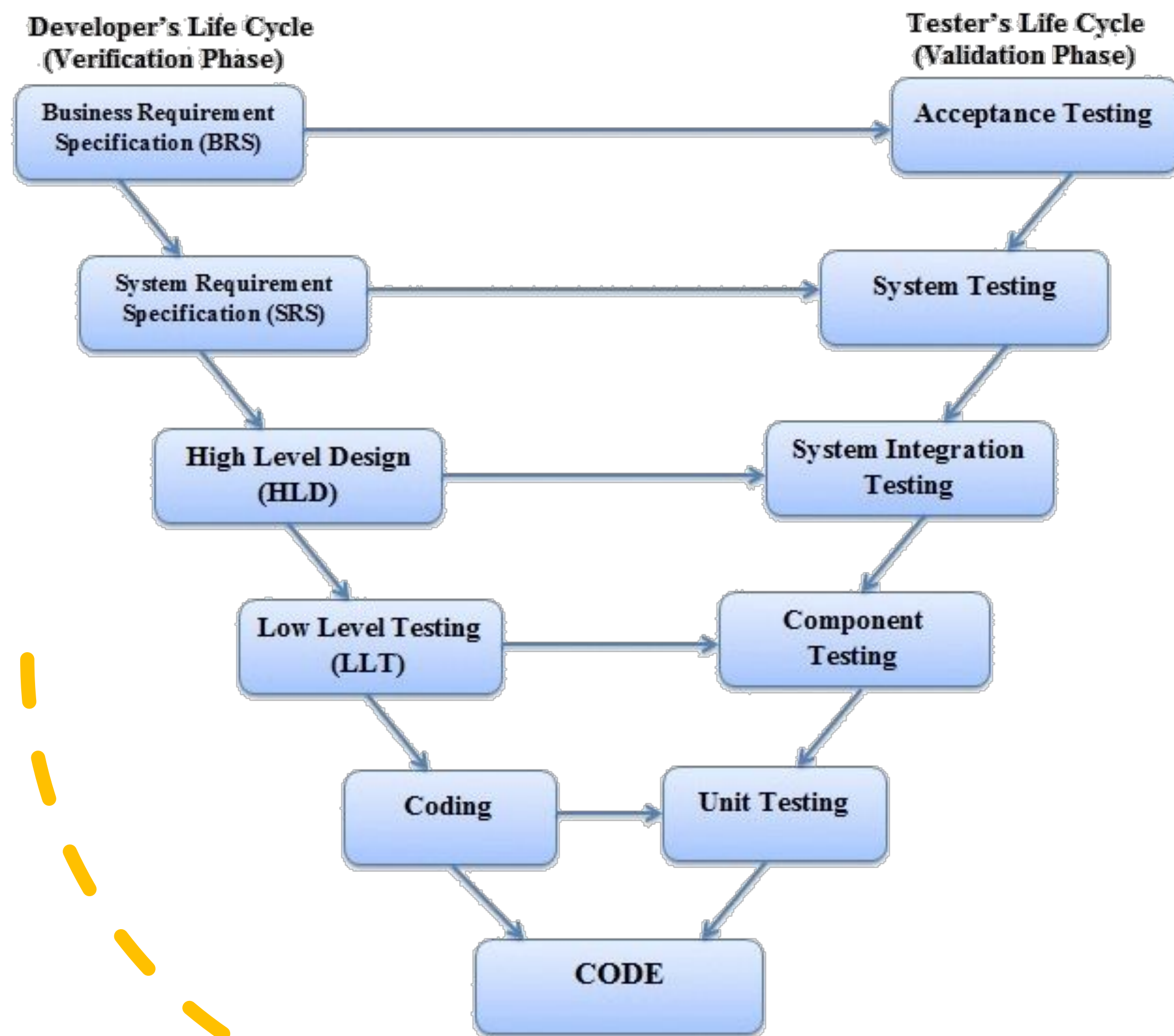
Analysis

- ☐ Difficulty in time management

V Model

- ❑ V-Model is mostly known as the validation and verification software development process model.
- ❑ It is one of the most know software development methodology.
- ❑ Although it is considered as an improvement to the waterfall model and it has some similarities as the process also based on sequential steps moving down in a linear way, it differs from the waterfall model as the steps move upwards after the coding phase to form the typical V shape.
- ❑ This V shape demonstrates the relationships between each phase of the development life cycle and its associated phase of testing. This means that any phase in the development process begins only if the previous phase is complete and has a correspondence related testing phase which is performed against this phase completion.
- ❑ Similar to the Waterfall model, the V-Model does not define the process to go back to the previous phase to handle changes in requirement.
- ❑ The technical aspect of the project cycle is considered as a V shape starting with the business needs on the upper left and ending with the user acceptance testing on the upper right.

V Model Cont.....



When to use,

- ☐ Where requirements are clearly defined and fixed.
- ☐ When ample technical resources are available with technical expertise.
- ☐ When the requirement is well defined and not ambiguous.
- ☐ The V-shaped model should be used for small to medium-sized projects where requirements are clearly defined and fixed.
- ☐ The V-shaped model should be chosen when sample technical resources are available with essential technical expertise.

V Model Cont....

Unit Testing

The Unit testing is the testing at the code level and helps eliminate issues at an early stage, mainly the developer is responsible to perform the unit test for his code while not all the defects cannot be discovered at the unit testing.

Functional Testing

Functional testing is associated with the low-level design phase which ensures that collections of codes and units are working together probably to execute new function or service.

Integration Testing

Integration testing is associated with the high-level design phase. Integration testing ensures the integration between all system modules after adding any new functions or updates.

System Testing

System testing is associated with the system requirements and design phase. It combines the software, hardware, and the integration of this system with the other external systems.

User Acceptance Testing

User Acceptance testing is associated with the business and operations analysis phase. The customer users are the main performers of this testing based on test cases and scenarios that cover the business requirements to ensure that they have delivered the right software as per the specifications.

V Model Cont....

Advantages

- ❑ This is a highly-disciplined model and Phases are completed one at a time.
- ❑ Works well for smaller projects where requirements are very well understood.
- ❑ Simple and easy to understand and use.
- ❑ Easy to manage due to the rigidity of the model.
- ❑ Each phase has specific deliverables and a review process.

V Model Cont....

Disadvantages

- ☐ High risk and uncertainty.
- ☐ Not a good model for complex and object- oriented projects.
- ☐ Poor model for long and ongoing projects.
- ☐ Not suitable for the projects where requirements are at a moderate to high risk of changing.
- ☐ Once an application is in the testing stage, it is difficult to go back and change a functionality.
- ☐ No working software is produced until late during the life cycle.

Recap from the lesson learnt

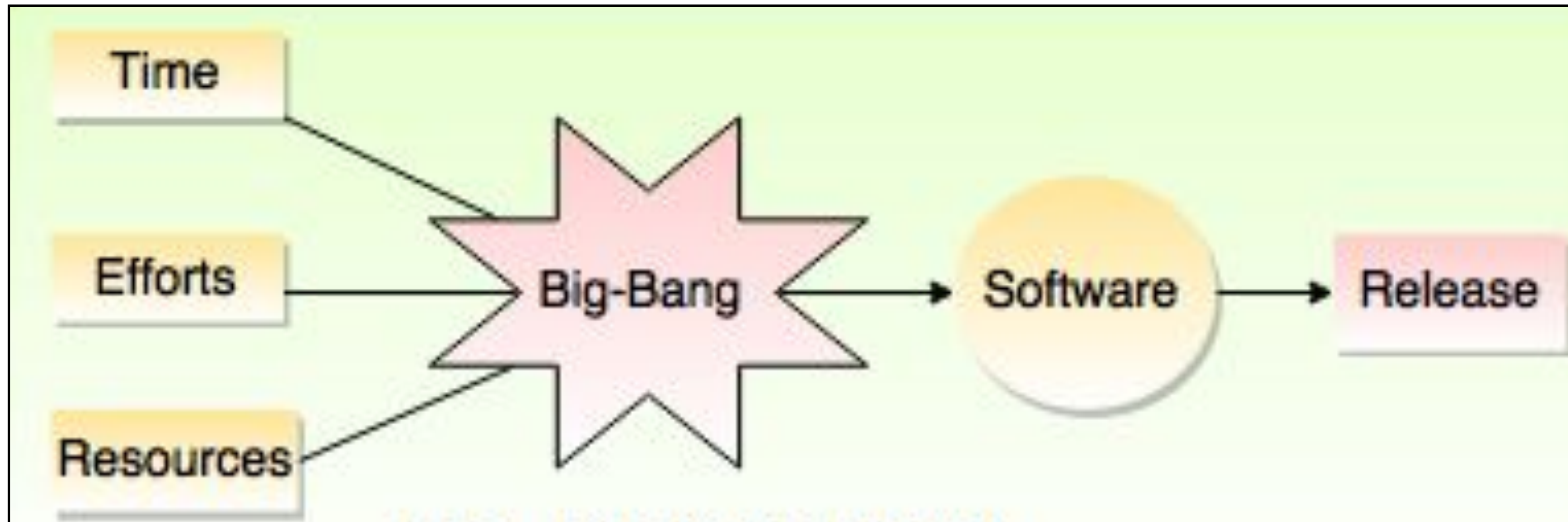
Mark the following True/False

1. The spiral model has four phases: Planning, risk analysis , Engineering and Evaluation. (True/False)
2. Spiral model is not suited to handle unknown risks after the project has started. (True/False)
3. We use a spiral model When the project is large. (True/False)
4. The Spiral Model is much more complex than other SDLC models.(True/False)
5. The V-model is a type of SDLC model where process executes in a sequential manner in V-shape. (True/False)
6. V-model is also known as the Vertical and Validate. (True/False)
7. V-Model is used for small projects where project requirements are clear.(True/False)

Big Bang Model

- ❑ The Big Bang Model is popular in situations or adopted when the project's stakeholders or customers are not certain regarding their wants and requirements.
- ❑ They have also haven't analyzed all the essential features and characteristics that they plan to incorporate into their system. In this model, all or most of the requirements are understood and put into practice as they begin developing the prototype.
- ❑ The primary target lingers in writing code as well as delivering the ultimate product to the customers. This model holds no planning or analysis, so this model involves many risks than other SDLC models.
- ❑ Though this model is moderately too simple to implement, there is still an imbalance in developing a product without proper analysis.


Big Bang Model Cont.....





Big Bang Model

Big Bang model can be phase out into the following points:

- ❑ In this development model, modules get integrated as all the individual modules are entirely built and are not integrated separately or individually.
 - ❑ Each module is tested independently to look for bugs or defects.
 - ❑ In case there is any bug or error in any module, the module is disintegrated, and the root cause of the problem is identified.
- 

Big Bang Model Cont.....

Advantages

- ☐ The advantage of Big Bang is that it's very simple and easy to implement.
- ☐ This model requires very little or no planning.
- ☐ There is no formal procedure are required before starting of any project so this model is easy to manage.
- ☐ It is ideal for repetitive or small projects with minimum risks.
- ☐ Very few resources required.
- ☐ Gives flexibility to developers

Big Bang Model Cont.....

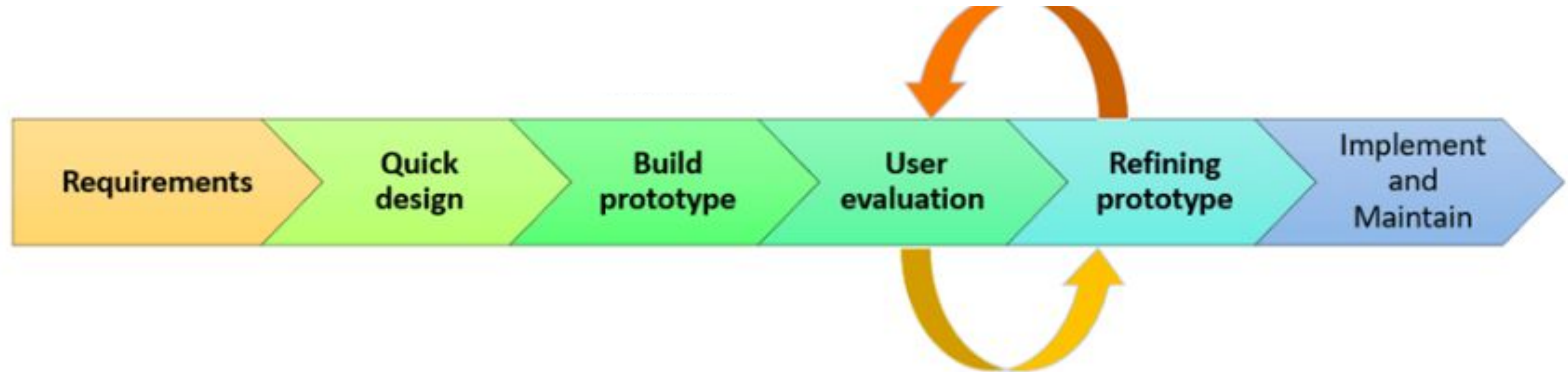
Disadvantages

- ❑ Due to there is no pre planning required before starting the project hence the Big Bang model is a very high risky model.
- ❑ In addition if changes in the requirements or misunderstood requirements may even lead to complete reversal or scraping of the project.

Prototype Model

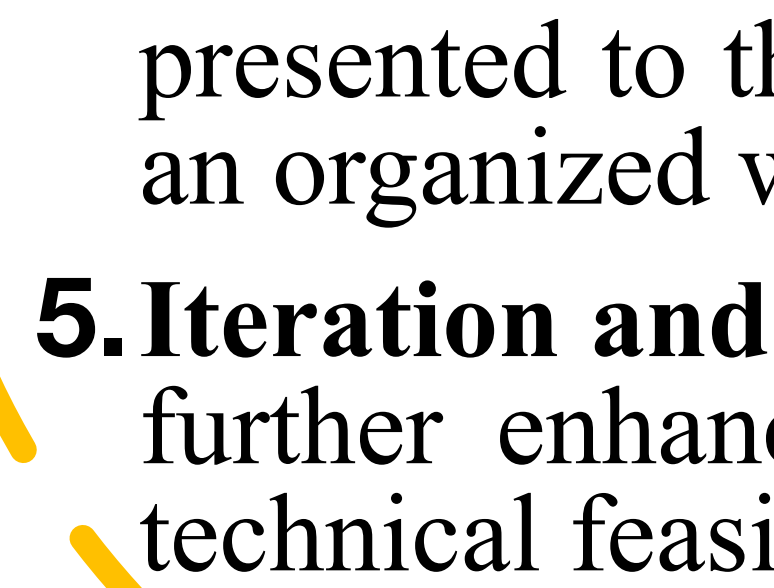
- ❑ The Prototyping SDLC approach is about creating a working prototype of the software product with limited functionality and then quickly turning the prototype into the complete product. The prototype may not contain the exact logic of the finished product.
- ❑ This software development life cycle approach is good for allowing the consumer to visualize the product. Gathering and analyzing feedback from customers helps the development team better understand customer requirements at the early stages of development.

Prototype Model Cont.....





Prototype Model Cont.....

- 1.Requirement Identification:** Here identification of product requirements is cleared in details. It is done through interview some product's future users and other members of the departments.
 - 2.Design Stage:** A first-round design is created in this stage for the new system.
 - 3.Build the Initial Prototype:** An initial prototype the target software is built from the original design. Working off all the product components may not be perfect or accurate. The first sample model is tailored as per the comments were given by the users and based on that the second prototype is built.
 - 4.Review of the Prototype:** After the product completes all the iterations of the update, it is presented to the customer or other stakeholders of the project. The response is accumulated in an organized way so that they can be used for further system enhancements.
 - 5.Iteration and Enhancement of Prototype:** Once the review of the product is done, it is set for further enhancement based on factors like - time, workforce as well as budget. Also, the technical feasibility of actual implementation is checked.
- 

Prototype Model Cont.....

Advantages

- ☐ Helps the developer to understand the certain and uncertain requirements of the customer.
- ☐ Easily realize the required modification before final implementation of the system.
- ☐ The efforts required in developing the final system is reduced.
- ☐ The customer does not have to wait for a long to see the working model of the final system.
- ☐ User-friendly. User satisfaction is achieved.

Prototype Model Cont.....

Disadvantages


- ☐ The prototype is developed in an artificial environment.
- ☐ User may never get satisfied as they are getting too much opportunity to change specify the changes.
- ☐ Number of iteration cannot be specify.
- ☐ Costly.
- ☐ Too much variation in requirements.
- ☐ Poor documentation for changing requirements.
- ☐ Optimal solution may not achieved.
- ☐ The customer satisfaction might lose if he/she not satisfied with the initial prototype.



Types of Prototyping used in the Software Industry

1. Throwaway Prototyping - Several prototypes are developed to understand the requirements of the customer and once the requirements are clear these prototypes are thrown away and the final system is developed using other process models.

2. Evolutionary Prototyping - Initially the prototype is developed by incorporating the specifications and then the developers proceed further to understand the unclear specification.



3. Incremental Prototyping - The final expected product is broken into different small pieces of prototypes and being developed individually. When all individual pieces are properly developed, the different prototypes are collectively merged into a single final product in their predefined order.



4. Extreme Prototyping - This method is mainly used for web development.

Recap from the lesson learnt

Mark the following True/False

1. The big bang model requires less planning, lots of coding or programming and lots of funds.(True/False)
2. Big Bang model This model is not appropriate for big or complex projects. (True/False)
3. Developers have huge flexibility in big bang model.(True/False)
4. The main purpose of the prototyping model is to satisfy the customer's needs. (True/False)
5. In prototype model, the customer suggests the modifications in the prototype.(True/False)
6. In throwaway prototyping, several prototypes are developed to understand the requirements of the customer. (True/False)
7. In prototype model, the efforts required in developing the final system can be reduced. (True/False)