

Latest Models in Software Testing

Submitted By-
Money Arora

C.Sc and IT Dept.

Software Testing

- Testing is the method or practice of finding all the errors or bugs while executing a program.
- Testing is the verification system which is used at the implementation phase.
- By using testing we can easily locate all the errors in the data prior to delivery to the end user.

Facts about Testing

- Finding all faults in software.
- Testing should be repeatable.
- Testing should be helpful for finding errors, not just detect their presence.
- It always provides optimal and appropriate results which are useful to the users.

Good Testing Practices

- Testability is a key objective in designing the software.
- Test case is supportive when it is appropriate for all valid as well as invalid input conditions.
- A test case is good for all cases if it has a high possibility of detecting an undiscovered consequence.
- The job of testing is always assigned to the best programmers.
- Testing phase is always applicable before the implementation phase.

Test Planning

A test plan includes following:-

- Test objectives
- Test strategies
- Logistics and plan
- All the information, procedure and predictable results
- Methods for managing harms

Types of Testing

- Unit Testing
- Integration Testing
- Validation Testing-
- It contains – Regression Testing, Alpha Testing, Beta Testing
- Acceptance Testing
- Black Box Testing
- White Box Testing

Unit Testing

- It includes logic and algorithms
- It includes local and global data structures.
- It contains some boundary conditions for the solution of problems
- It also contains some error handling techniques.
- Developers play a very important role in unit testing.
- The main goal of unit testing is to check that subsystems are correctly coded and carries out all the functionality

Integration Testing

- It includes all the groups of subsystems and in other words the whole entire system
- It is carried out by Developers
- The main goal of integration testing is properly examine the interface among all the subsystems
- It is suitable for handling all the timing problems(in real-time systems)

Types of Integration Testing

There are two types of Integration Testing which are as follows:-

- **Top down Integration-** In this integration, main control module is divided into very small modules so that errors are easily detected.
- **Bottom-up Integration-** In this integration, the integration begins with the lowest-level modules which are combined into clusters that perform a specific software sub function.

Validation Testing

- It determines whether the software fulfills all the requirements which are defined in the SRS
- **Regression Testing- 1)** Regression Testing is useful to examine whether the software still fulfills all the requirements after modifications and changes in the software.
2) It contains various validation tests and also developing new tests for optimal results.
- **Alpha and Beta Testing-1)** its main task is to focus on the specific and appropriate test scenarios for executing the problems.
2) It finds out all the defects that are discovering while execution.

Acceptance Testing

- It is similar to validation testing except that in which customers are present or directly involved.
- Usually customers are also developed the tests in acceptance testing.
- It evaluates the system delivery by developers.
- It involves executing various transactions on the trial basis on different sites.

Black Box Testing

- The main focus of black box testing is on the input/output behavior. For any given Input, we can easily find out the optimal output.
- Equivalence partitioning helps to reduce the number of test cases.
- In black box testing, input is valid for different range of values. We can make the test cases for different equivalence classes.

White Box Testing

- Every single statement in the component is executed at least once.
- It contains following types:-
- **Statement Testing:** - It test only single statement.
- **Loop Testing :** -It test large number of statements. In which loop is executed exactly once and more than once.
- **Path Testing:-** All the paths in the program are executed by the path testing.
- **Branch Testing:-** Each possible outcome from a given condition is tested at least once in branch testing.

Latest Models in Software Testing

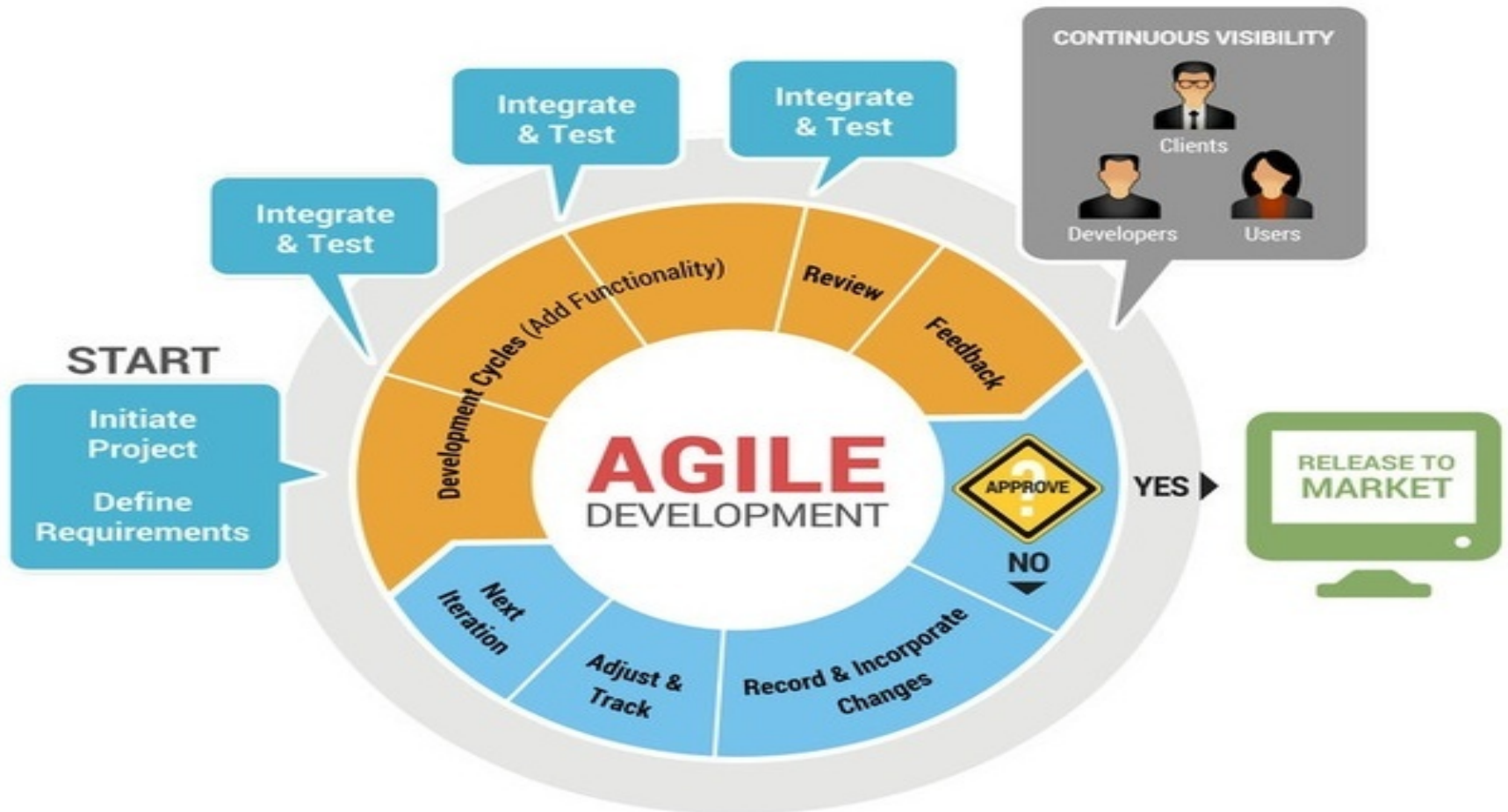
- Agile Software Development Methodology
- Rapid Application Development Methodology
- Dynamic System Development Model Methodology
- Extreme Programming Methodology
- Feature Driven Development Methodology
- Joint Application Development Methodology
- Lean Software Development Methodology
- Rational Unified Process Methodology
- Scrum Development Methodology

Agile Software Development

Methodology

- Agile methodology is a technique that helpful for the development and testing all over the software development lifecycle of the project.
- Unlike the Waterfall model, all the development and testing activities are synchronized in it.
- It stress on four main facts which are as follows:
 - Individual and team interactions
 - Customer Collaboration
 - Working over Comprehensive documentation
- It is considered as a unstructured compared to Waterfall model.
- Implementation of small projects can be done very quickly.

Agile Software Development Methodology

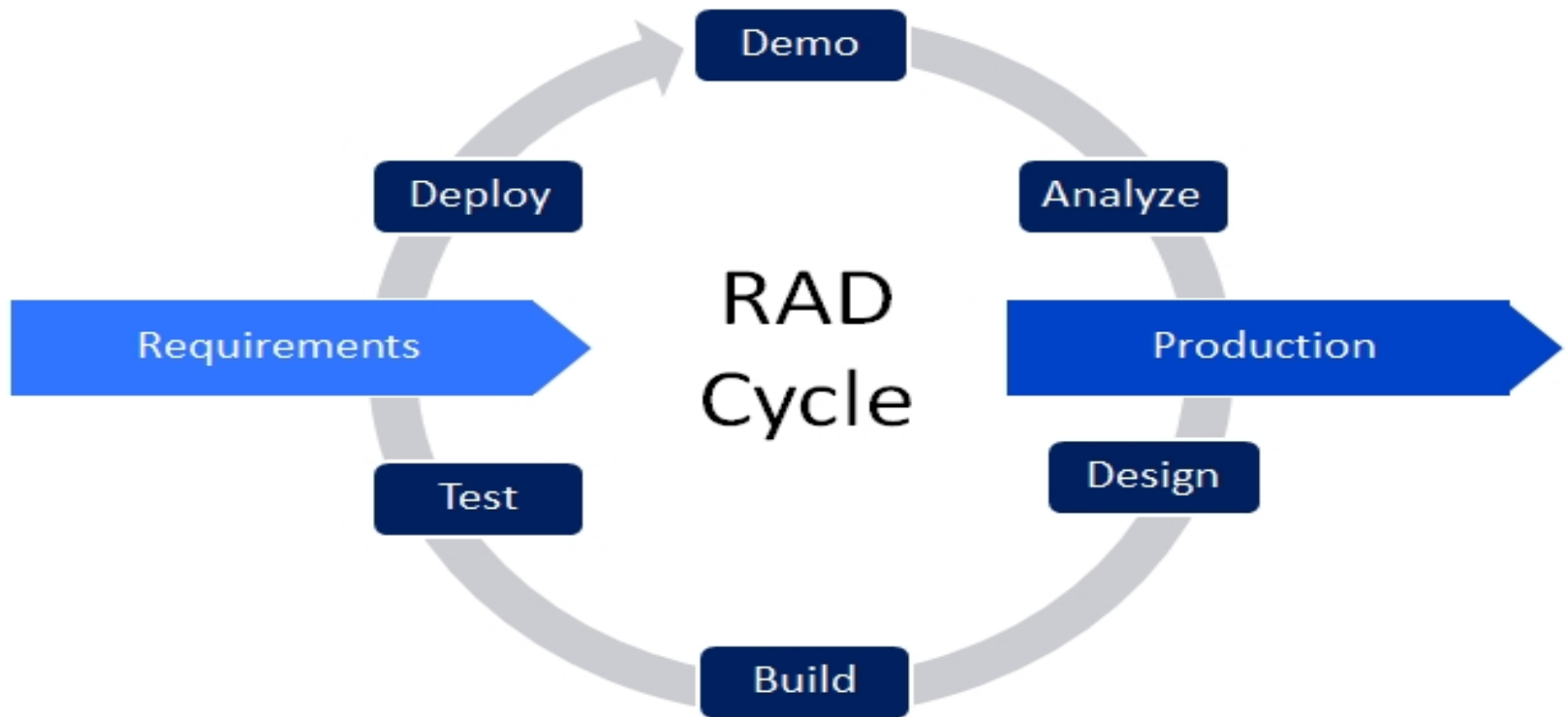


Rapid Application Development

Methodology

- It is an alternative to the conventional waterfall model of software development.
- It's main focus on the process rather than on the planning.
- It is useful for adapting and adjusting requirements for the software development.
- In place of design specifications, prototypes are used in this methodology.
- Graphical user interface builders are sometimes treated as a RAD tool.
- Advantages of RAD are as: Better Quality, Risk Control
- Disadvantages of RAD are as: The risk of new approach, requires time for scarce resources, Less control, Poor design.

Rapid Application Development Methodology



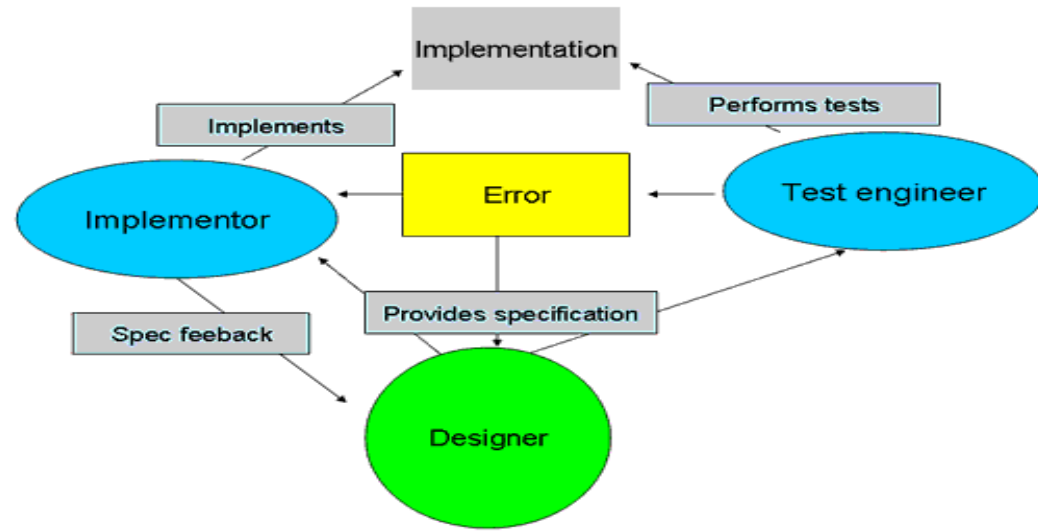
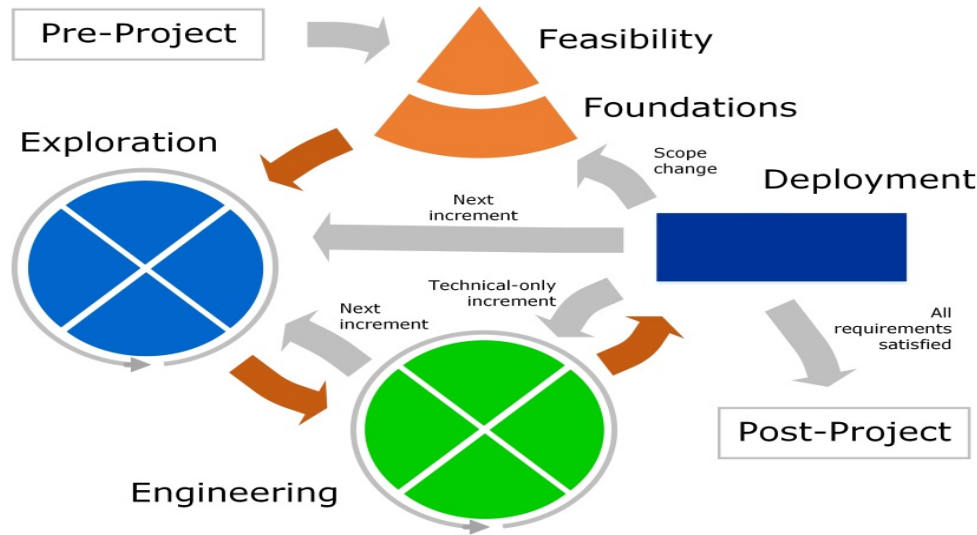
Dynamic System Development

Model Methodology

- It is one of the useful software development methods which are invented in 1994.
- It also provides some beneficial discipline to RAD method and became a generic approach to project management.
- It includes the continuous user/customer involvement which is a very important principle of agile development.
- It is one of the agile methods which are used as developing software and non-IT solutions.

Dynamic System Development Model

Methodology

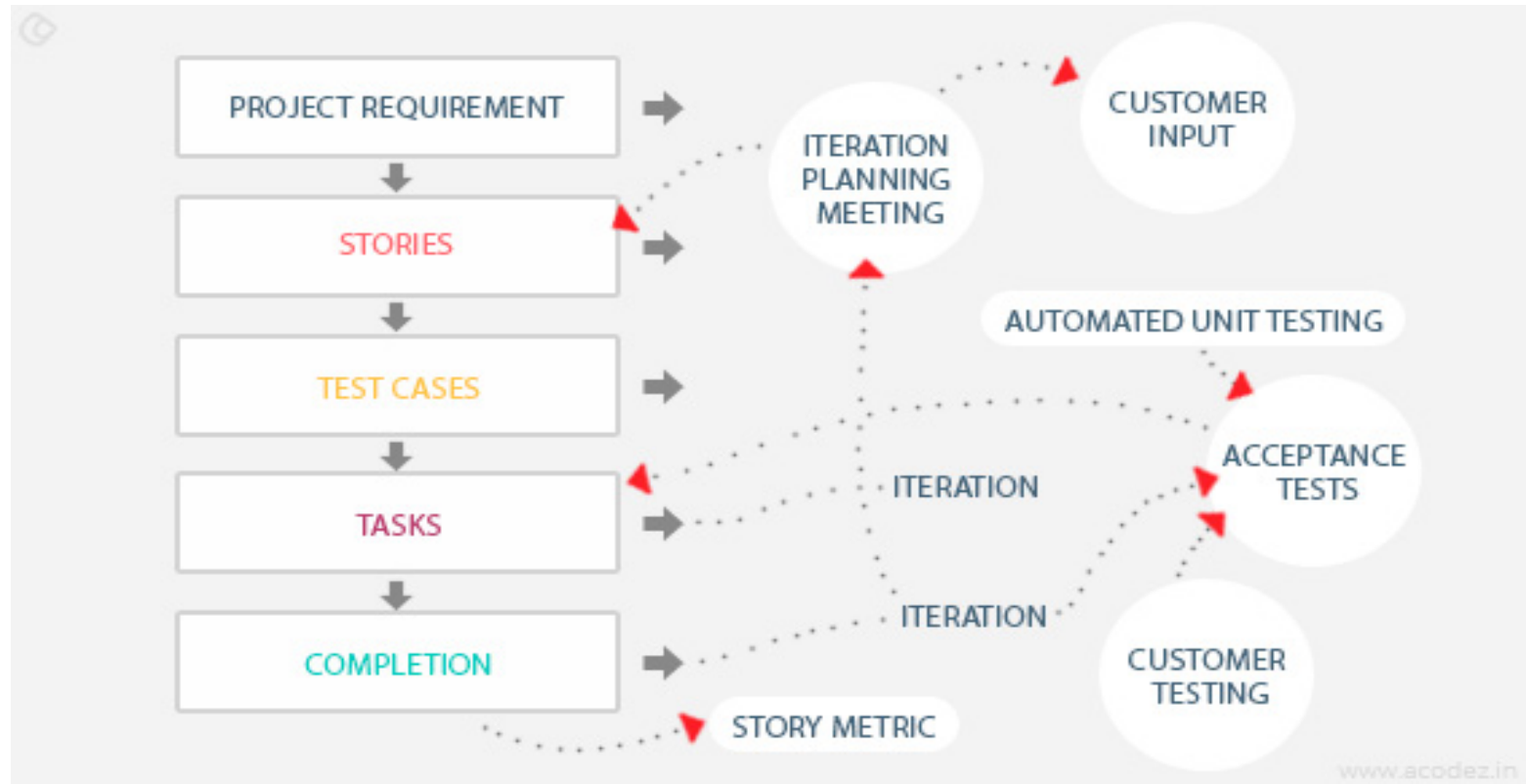


Extreme Programming

Methodology

- It is a lightweight, efficient, low-risk, flexible and technique for developing software.
- It is one of the agile software development methods.
- It provides all the principles to the users those are involved in developing the software and fulfills its requirements.
- Communication, Simplicity, Feedback, Courage and Respect are the basis of the extreme programming.

Extreme Programming Methodology

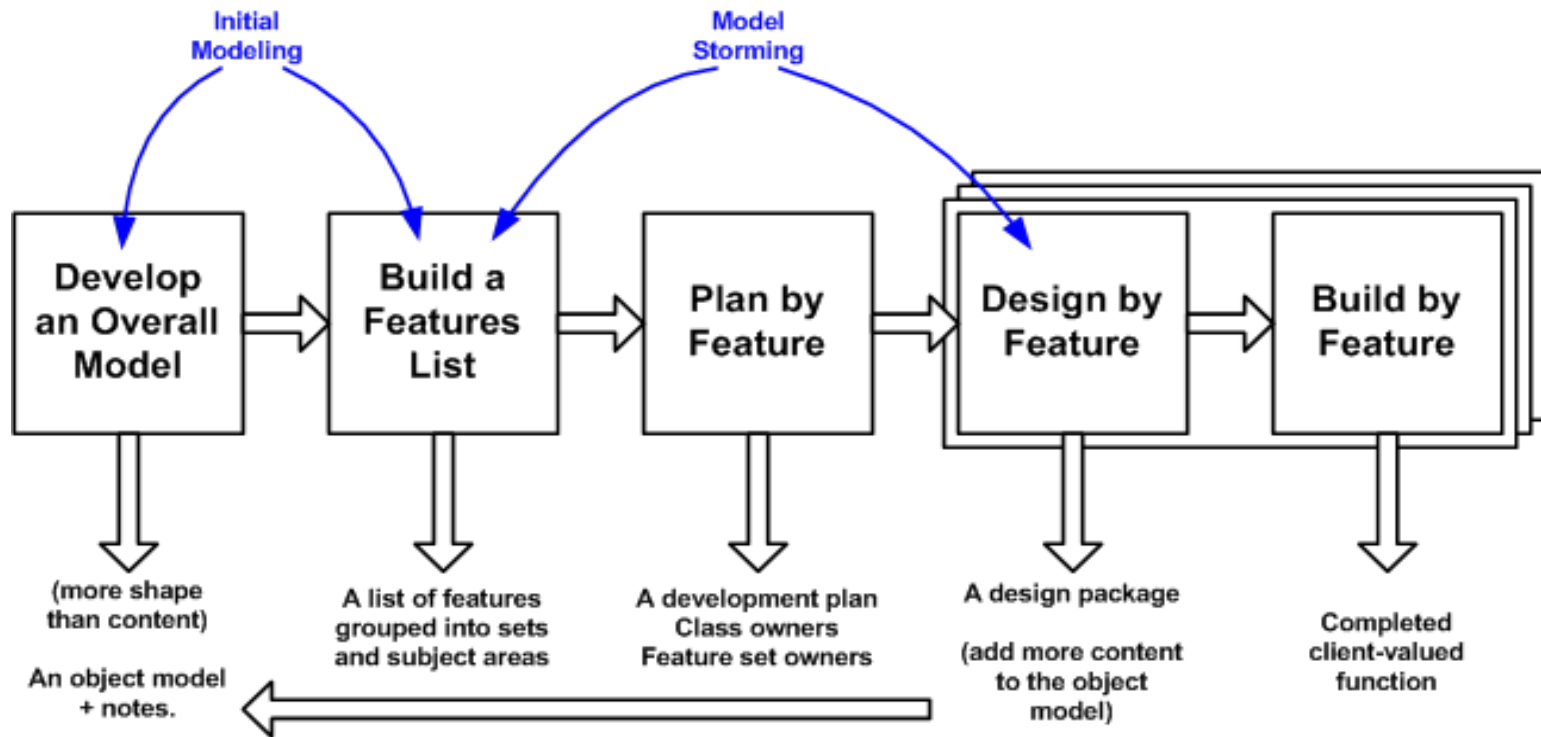


Feature Driven Development

Methodology

- This software development process is very incremental and iterative.
- It is one of the lightweight or agile methods for enhancing the software.
- Its main aim is to provide tangible software in a timely manner.
- It is developed by the Jeff De Luca in 1997.
- FDD tools, CASE Spec and FDD Viewer are the effective tools used in this methodology.

Feature Driven Development Methodology

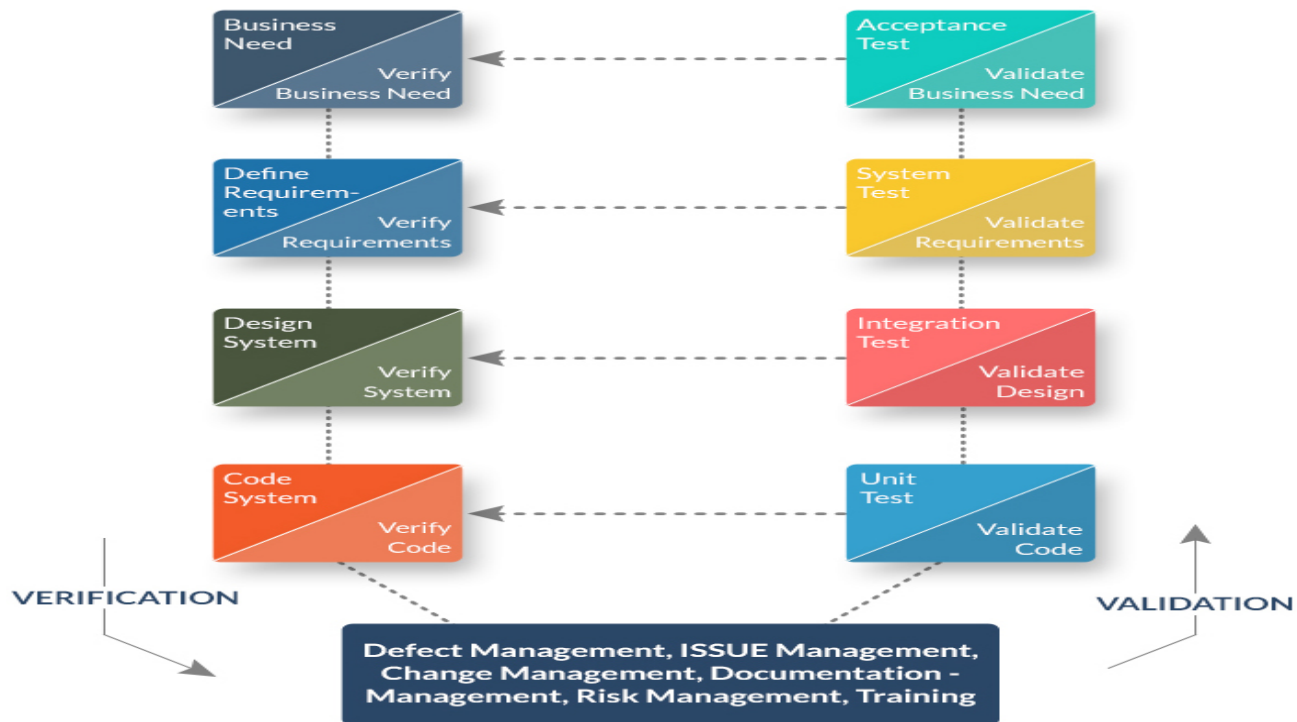


Joint Application Development

Methodology

- It is applicable to any development process.
- It is used for designing a computer-based system.
- The main advantage of this method is to continuously conversation with the users and different designers.
-
- Lean Software Development Methodology
- Lean Software Development is a translation of lean manufacturing principles and practices to the software development domain.
- It is adapted from the Toyota Production System.
- Its main manufacturing principles are:
 - Eliminate waste
 - Amplify learning
 - Empower the team
 - Deliver as fast as possible
 - Decide as late as possible.

Joint Application Development Methodology



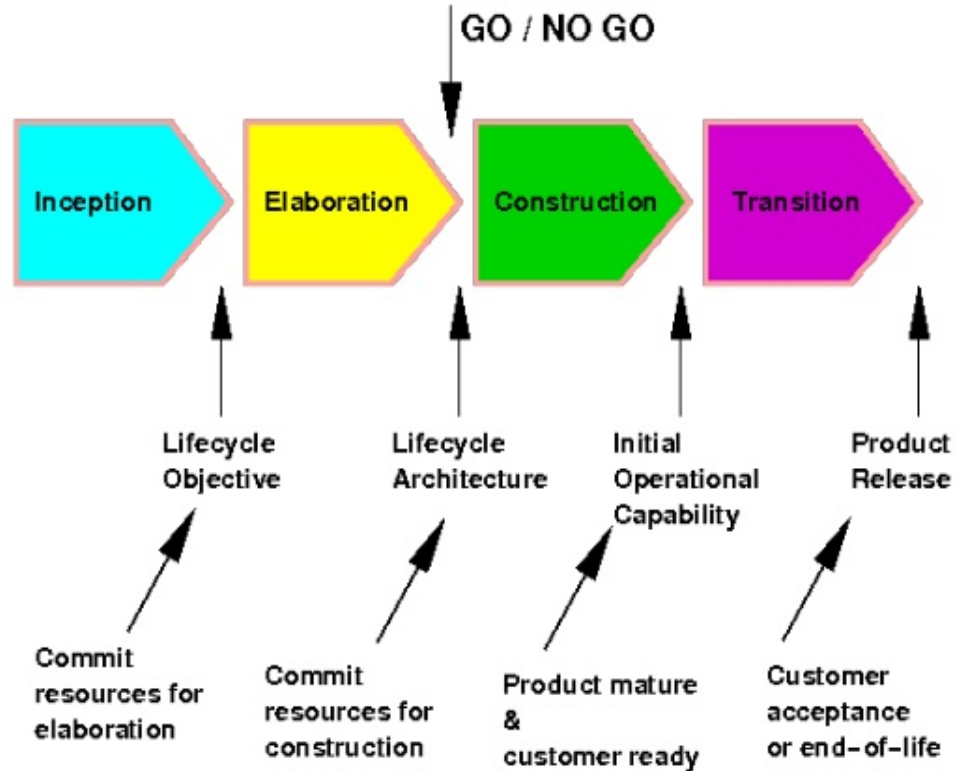
Rational Unified Process

Methodology

- This software development process is invented by the Rational Software Corporation, a part of IBM since 2003.
- It is flexible process for development organizations and software project teams.
- It includes six engineering disciplines like business modeling, requirements, analysis and design, implementation, test and deployment etc.
- It includes supporting disciplines like configuration and change management, project management and environment.

Rational Unified Process Methodology

PHASES OF RUP



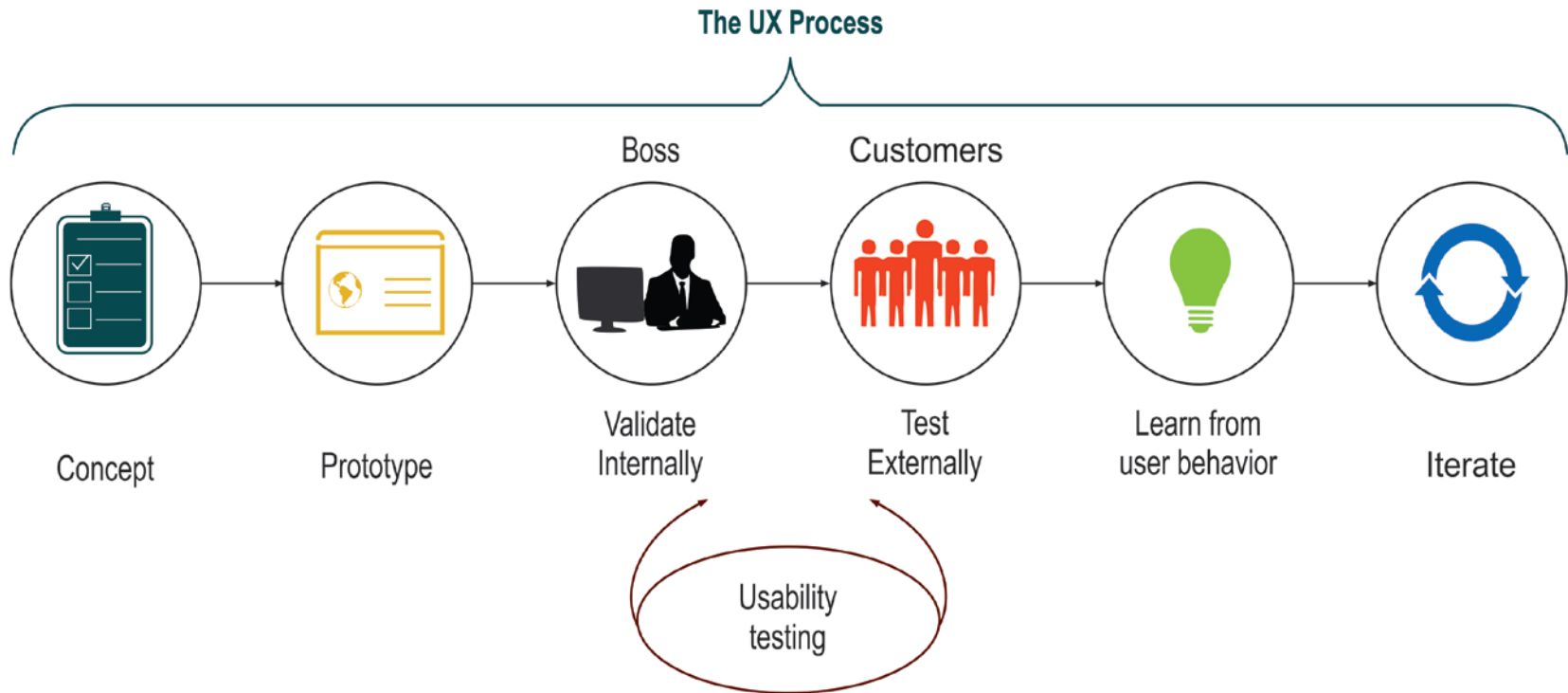
Lean Software Development

Methodology

- Lean Software Development is a translation of lean manufacturing principles and practices to the software development domain.
- It is adapted from the Toyota Production System.
- Its main manufacturing principles are:
 - Eliminate waste
 - Amplify learning
 - Empower the team
 - Deliver as fast as possible
 - Decide as late as possible

Lean Software Development Methodology

Lean Development (LD)



Scrum Development Methodology

- Scrum is an agile development method.
- The main focus for developing any software is a resource planning.
- It is efficient method for developing software.

Scrum Development Methodology

