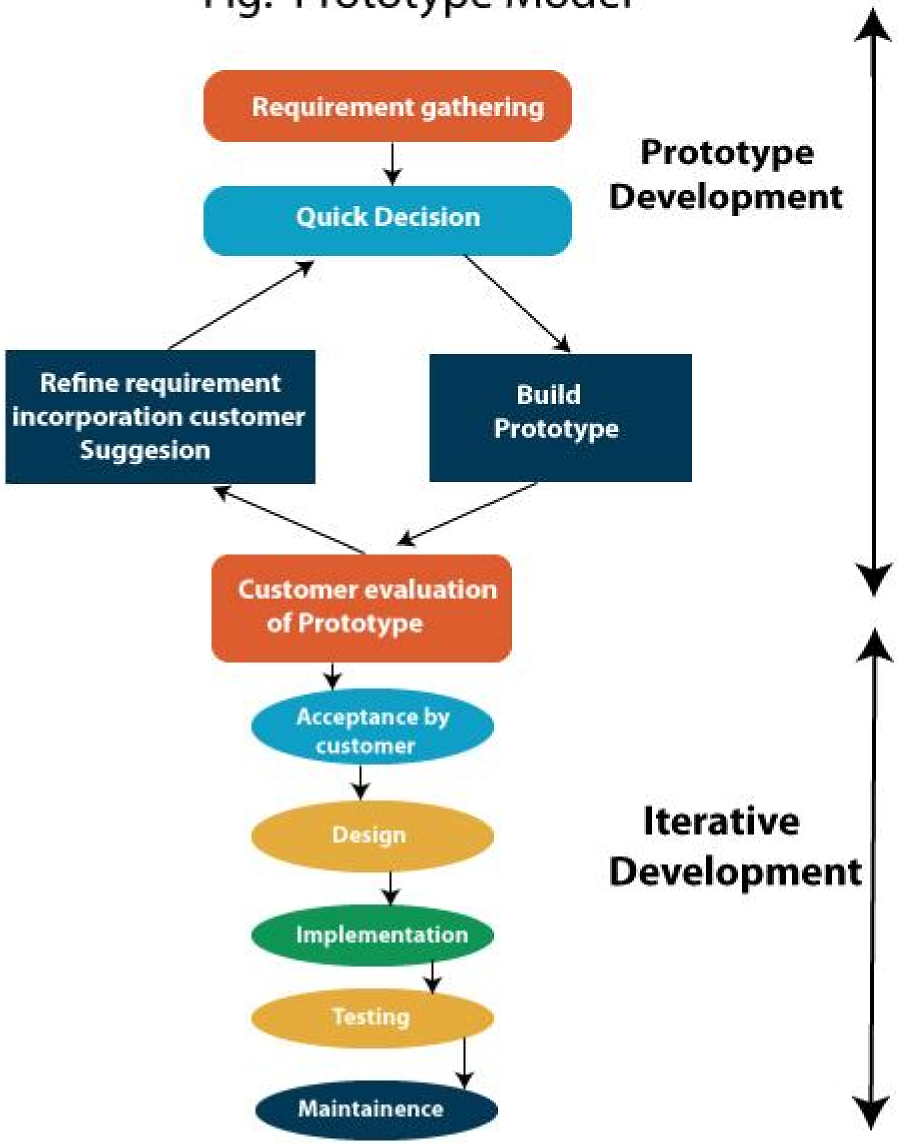
PROTOTYPE MODEL

The prototype model requires that before carrying out the development of actual software, a working prototype of the system should be built. A prototype is a toy implementation of the system. A prototype usually turns out to be a very crude version of the actual system, possible exhibiting limited functional capabilities, low reliability, and inefficient performance as compared to actual software. In many instances, the client only has a general view of what is expected from the software product. In such a scenario where there is an absence of detailed information regarding the input to the system, the processing needs, and the output requirement, the prototyping model may be employed.

Fig: Prototype Model



Steps of Prototype Model

1. Requirement Gathering and Analyst
2. Quick Decision
3. Build a Prototype
4. Assessment or User Evaluation
5. Prototype Refinement
6. Engineer Product

Advantage of Prototype Model

1. Reduce the risk of incorrect user requirement
2. Good where requirement are changing/uncommitted
3. Regular visible process aids management
4. Support early product marketing
5. Reduce Maintenance cost.
6. Errors can be detected much earlier as the system is made side by side.

Disadvantage of Prototype Model

1. An unstable/badly implemented prototype often becomes the final product.
2. Require extensive customer collaboration
   - Costs customer money
   - Needs committed customer
   - Difficult to finish if customer withdraw
   - May be too customer specific, no broad market
3. Difficult to know how long the project will last.
4. Easy to fall back into the code and fix without proper requirement analysis, design, customer evaluation, and feedback.
5. Prototyping tools are expensive.
6. Special tools & techniques are required to build a prototype.
7. It is a time-consuming process.

Hybrid Model

The hybrid model is the combination of two or more primary (traditional) models and modifies them as per the business requirements. This model is dependent on the other SDLC models, such as spiral, V and V, and prototype models. The hybrid model is mainly used for small, medium, and large projects. It focuses on the risk management of the product.

We go for the hybrid model whenever we want to obtain the features of two models in a single model. And when the model is dependent and the customer is new to the industry.

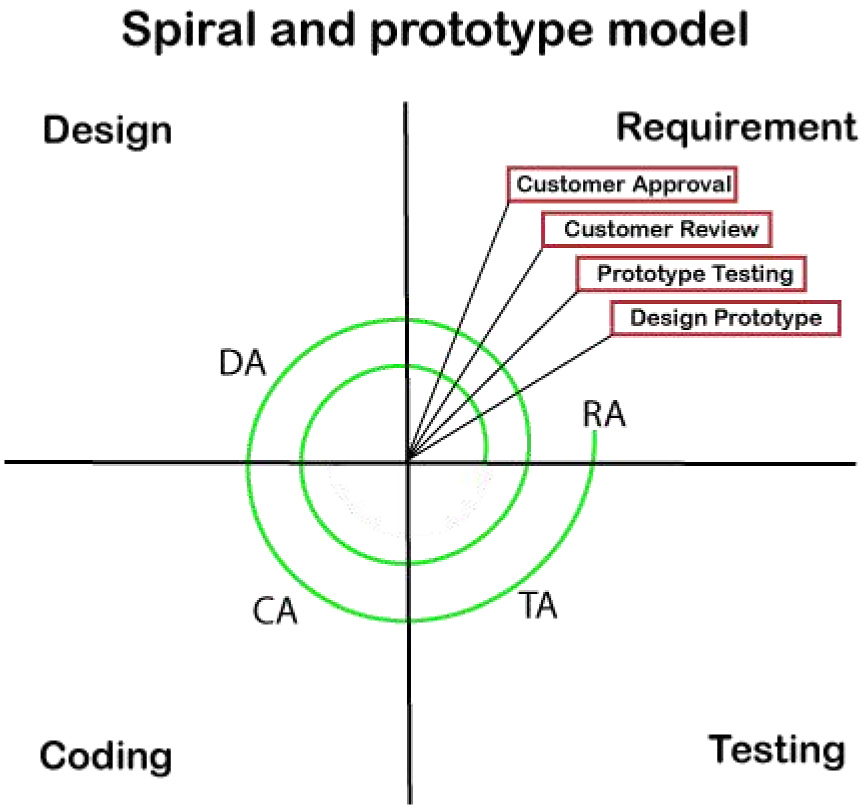
The most commonly used combination of two models is as follows:

- **Spiral and prototype**
- **V & V and Prototype**

Spiral & Prototype

The spiral and prototype model is used for the below conditions:

- We go for spiral and prototype models whenever there is dependency.
- When the customer gives requirements in stages, and we develop the product in stages using this model.
- When the customer is very new to the software industry and not clear about the requirements.
- When the developers are new to particular software.
-



V & V and prototype Model

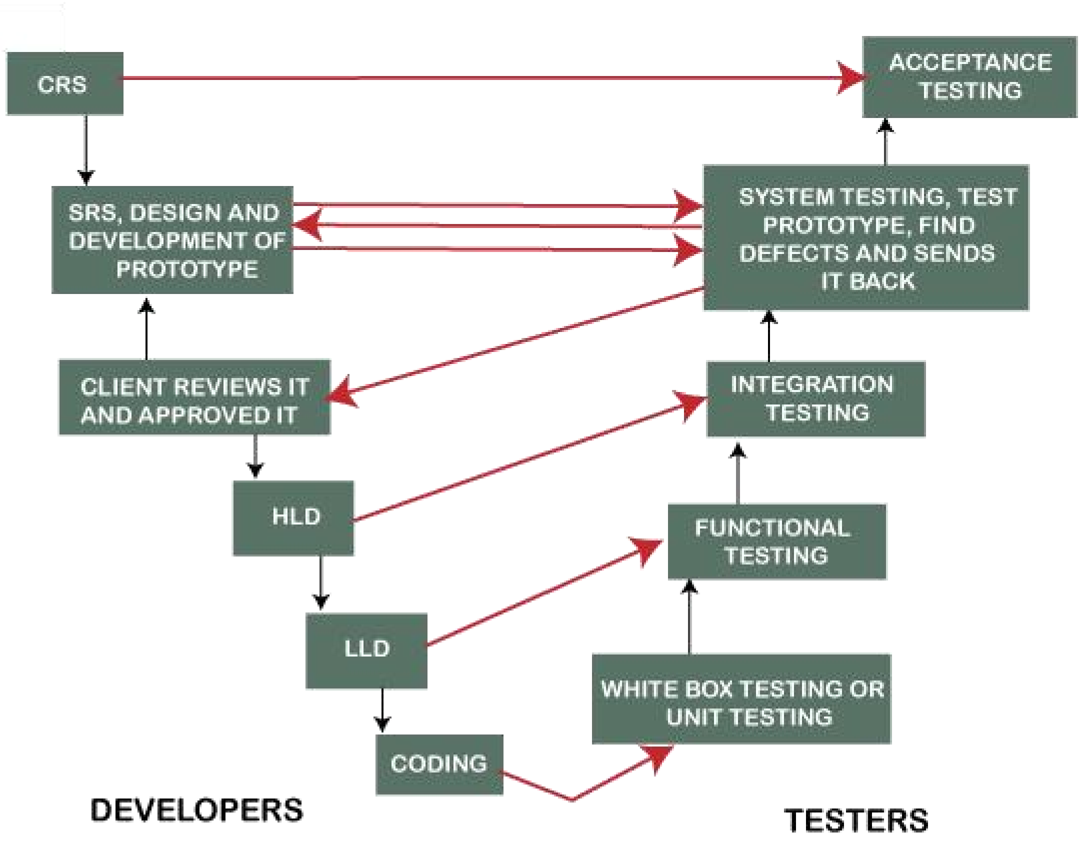
We go for this model for the following reasons:

- When the customer and developers are both new to the industry.
- When the clients are expecting a very high-quality product within the required time because every phase is tested, and the developer and testing team are working paralelly.

V & V and Prototype model process

In the hybrid model, the testing team is involved to test the Prototype.In this, testing will start from the early stage of product development, which avoids the downward flow of bugs, which helps us to reduce the re-work.

V & V and prototype model


CRS
ACCEPTANCE TESTING
SRS, DESIGN AND DEVELOPMENT OF PROTOTYPE
SYSTEM TESTING, TEST PROTOTYPE, FIND DEFECTS AND SENDS IT BACK
CLIENT REVIEWS IT AND APPROVED IT
INTEGRATION TESTING
HLD
FUNCTIONAL TESTING
LLD
WHITE BOX TESTING OR UNIT TESTING
CODING
DEVELOPERS
TESTERS

Advantages

- The hybrid model is highly flexible.
- In this model, the customer rejection is less because of the Prototype.
- It is easy to implement because it has the flexibility of synchronization.
- It is easy to use and apply, especially with small and medium projects.
- In this, the development process will be smooth and quick because here we follow only the relevant process cycles.

Disadvantages

- Every hybrid model is different from each other.
- It does not follow the usual standards.

GRAY BOX TESTING

Gray-box testing is a combination of white-box testing and black-box testing. The aim of this testing is to search for the defects, if any, due to improper structure or improper usage of applications.

- It provides combined benefits of both black box testing and white box testing both
- It combines the input of developers as well as testers and improves overall product quality
- It reduces the overhead of long process of testing functional and non-functional types
- It gives enough free time for a developer to fix defects
- Testing is done from the user point of view rather than a designer point of view

Techniques

gray-box testing as involving inputs and outputs, but test design is educated by information about the code or the program operation of a kind that would normally be out of view of the tester". Gray-box testing techniques are:

- Matrix Testing: states the status report of the project.
- Regression testing: it implies rerunning of the test cases if new changes are made.
- Pattern Testing: verify the good application for its design or architecture and patterns.
- Orthogonal array testing: used as subset of all possible combination.

Selenium Grid

Selenium Grid is a part of the Selenium Suite that specializes in running multiple tests across different browsers, operating systems, and machines in parallel. It is achieved by routing the commands of remote browser instances where a server acts as a hub. A user needs to configure the remote server in order to execute the tests.

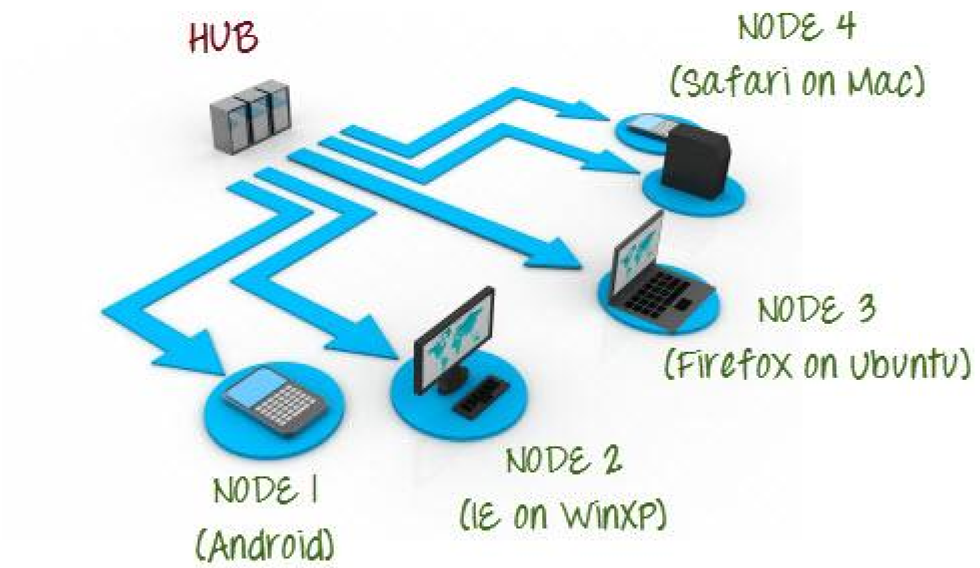
Hub

- The hub is the central point where you load your tests into.
- There should only be one hub in a grid.
- The hub is launched only on a single machine, say, a computer whose OS is Windows 7 and whose browser is IE.
- The machine containing the hub is where the tests will be run, but you will see the browser being automated on the node.

Nodes

- Nodes are the Selenium instances that will execute the tests that you loaded on the hub.
- There can be one or more nodes in a grid.
- Nodes can be launched on multiple machines with different platforms and browsers.

- The machines running the nodes need not be the same platform as that of the hub.
- Selenium Grid uses a hub-node concept where you only run the test on a single machine called a **hub**, but the execution will be done by different machines called **nodes**



How to Set Up Selenium Grid?

In this section, you will use 2 machines. The first machine will be the system that will run the hub while the other machine will run a node. For simplicity, let us call the machine where the hub runs as "Machine A" while the machine where the node runs will be "Machine B." It is also important to note their IP addresses. Let us say that Machine A has an IP address of 192.168.1.3 while Machine B has an IP of 192.168.1.4.

Step 1

Download the Selenium Server from here.

SRS VS BRS

BRS (Business Requirement Specification)	SRS (System Requirement Specification)
It describes at very high level the functional specifications of the software	It describes at a high level , the functional and technical specification of the software
It is a formal document describing about the requirement provided by client (written, verbal)	It specifies the functional and non-functional requirements of the software to be developed
Usually its created by the Business Analyst who interacts with clients	Usually its created by the System Architect who is an technical expert . Though in smaller companies the BA will create SRS as well. Some companies do not create SRS altogether. Their BRS is detailed enough to be used as SRS as well.
It is derived from client interaction and requirements	It is derived from the BRS

Comparison between Requirement and Specification

They outline "what" the software must do	They outline "how" the software will be created

They outline the software from the end-user , business and stakeholder perspective.	They outline the software from the technical team perspective.