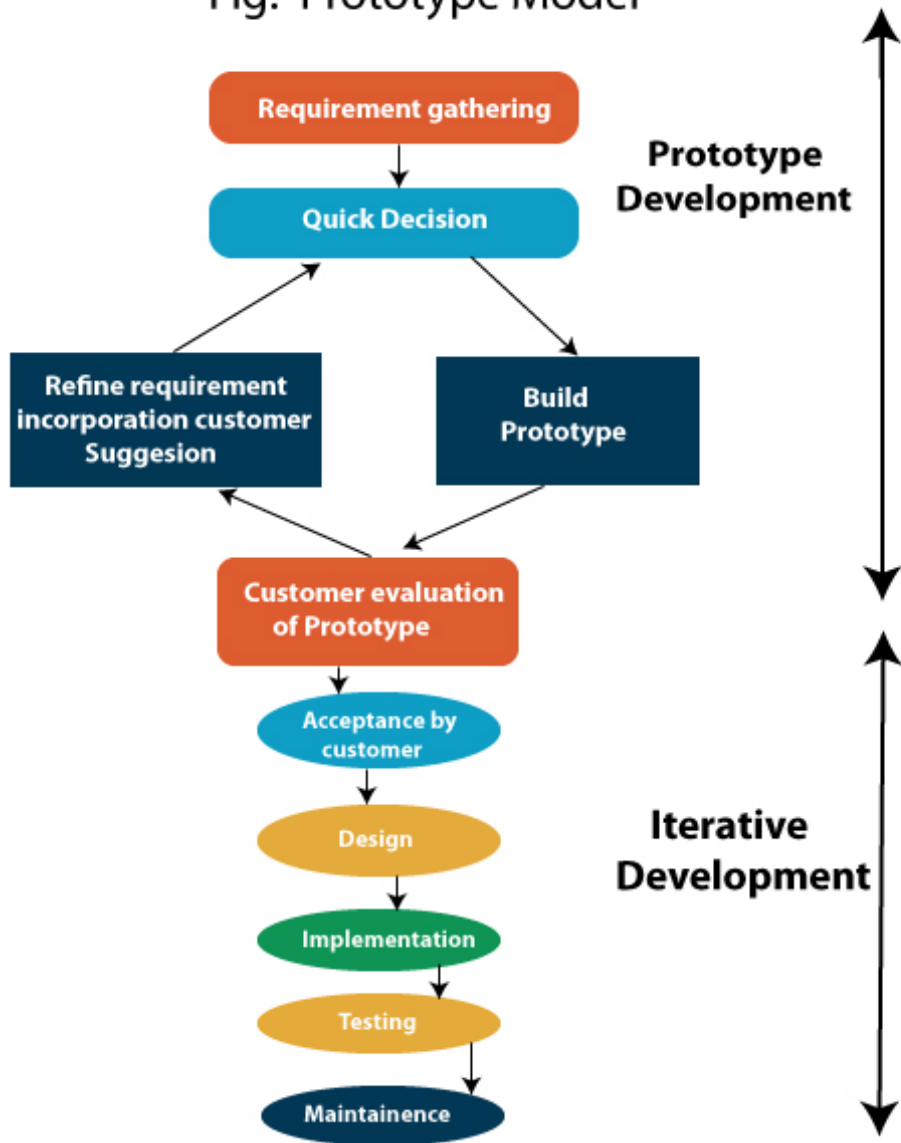




Prototype Model

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, possibly 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.

Evolutionary Process Model

Evolutionary process model resembles the iterative enhancement model. The same phases are defined for the waterfall model occurs here in a cyclical fashion. This model differs from the iterative enhancement model in the sense that this does not require a useful product at the end of each cycle. In evolutionary development, requirements are implemented by category rather than by priority.

For example, in a simple database application, one cycle might implement the graphical user Interface (GUI), another file manipulation, another queries and another updates. All four cycles must complete before there is a working product available. GUI allows the users to interact with the system, file manipulation allow the data to be saved and retrieved, queries allow user to get out of the system, and updates allows users to put data into the system.

Benefits of Evolutionary Process Model

Use of EVO brings a significant reduction in risk for software projects.

EVO can reduce costs by providing a structured, disciplined avenue for experimentation.

EVO allows the marketing department access to early deliveries, facilitating the development of documentation and demonstration.

Better fit the product to user needs and market requirements.

Manage project risk with the definition of early cycle content.

Uncover key issues early and focus attention appropriately.

Increase the opportunity to hit market windows.

Accelerate sales cycles with early customer exposure.

Increase management visibility of project progress.

Increase product team productivity and motivations.

← Prev

Next →

 Youtube For Videos Join Our Youtube Channel: [Join Now](#)


Feedback

- Send your Feedback to feedback@javatpoint.com

Help Others, Please Share





Learn Latest Tutorials


 Splunk tutorial
Splunk


 SPSS tutorial
SPSS


 Swagger tutorial
Swagger


 T-SQL tutorial
Transact-SQL


 Tumblr tutorial
Tumblr

 React tutorial
ReactJS

 Regex tutorial
Regex

 Reinforcement learning tutorial
Reinforcement Learning


 R Programming tutorial
R Programming


 RxJS tutorial
RxJS

 React Native tutorial
React Native

 Python Design Patterns
Python Design Patterns


 Python Pillow
tutorial
Python Pillow


 Python Turtle
tutorial
Python Turtle

 Keras tutorial
Keras

Preparation

 Aptitude
Aptitude

 Logical
Reasoning
Reasoning

 Verbal Ability
Verbal Ability

 Interview
Questions
Interview Questions

 Company
Interview
Questions
Company Questions

Trending Technologies



Artificial
Intelligence
Tutorial

Artificial
Intelligence



AWS Tutorial
AWS



Selenium
tutorial
Selenium



Cloud
Computing
tutorial
Cloud Computing



Hadoop tutorial
Hadoop



ReactJS
Tutorial
ReactJS



Data Science
Tutorial
Data Science



Angular 7
Tutorial
Angular 7



Blockchain
Tutorial
Blockchain



Git Tutorial
Git




Machine
Learning Tutorial
Machine Learning





DevOps
Tutorial
DevOps

B.Tech / MCA

 DBMS tutorial
DBMS

 Data Structures
tutorial
Data Structures


 DAA tutorial
DAA

 Operating
System tutorial
Operating System


 Computer
Network tutorial
Computer Network


 Compiler
Design tutorial
Compiler Design


 Computer
Organization and
Architecture
Computer
Organization

 Discrete
Mathematics
Tutorial
Discrete
Mathematics


 Ethical Hacking
Tutorial
Ethical Hacking


 Computer
Graphics Tutorial
Computer Graphics


 Software
Engineering
Tutorial
Software
Engineering

 html tutorial
Web Technology

 Cyber Security
tutorial
Cyber Security


 Automata
Tutorial
Automata


 C Language
tutorial
C Programming


 C++ tutorial
C++

 Java tutorial
Java


 .Net
Framework
tutorial
.Net

 Python tutorial
Python

 List of
Programs
Programs

 Control
Systems tutorial
Control System

 Data Mining
Tutorial
Data Mining

 Data
Warehouse
Tutorial
Data Warehouse