

3.6.3 Software Development Models

The following table explains some of the different standardized models that developers use when developing new software:

Model	Description
Ad Hoc	<u>Ad hoc development is usually chaotic in nature. It is when the most qualified developers are given a project without a consistent team, funding, or a schedule. The outcome of the product lies solely in the hands of the developer, rather than an organization. Ad-hoc should be a last alternative when choosing a development method.</u>
Waterfall Planning	<p><u>The waterfall planning model is sequential in its layout. Each phase contains a series of instructions that must be executed and documented before the next phase can begin. This is the most commonly used model, though it may not always be optimal for large and complex projects. There are a few variations of the names of its phases, but it is mainly a sequence of events:</u></p> <ul style="list-style-type: none"> Requirements - the requirements are gathered from the client, user, or stakeholder. Design - the software is documented, diagramed, and designed. Implementation - all coding is done and the software is ready. Testing - client tests and accepts the software, as requested. Deployment - the software goes into production. Maintenance - any bugs or enhancements are addressed.
Structured Programming	<u>Structured programming development is a method used by programmers that allows for optimal control over coherence, security, accuracy, and comprehensibility. It uses layering, modularity, and segmenting in its method and usually requires processes to be defined and each sequence or phase to be reviewed. This is one of the most widely used development models.</u>
Prototype	<p><u>The prototype model is a type of iterative development that was made to combat the weaknesses of waterfall-based models. In the prototype model, a small segment of the code is prototyped, then tested and refined using four steps:</u></p> <ol style="list-style-type: none"> 1. Definition of initial concept 2. Implementation of initial prototype 3. Refinement of prototype until functional 4. Complete and release the final version
Spiral	<u>The spiral model is a mix of the waterfall model and the prototype model in which a prototype is developed and tested using the waterfall method. Considerations for improvements are implemented from the center outward, like a spiral. Additionally, the spiral method includes risk assessment. During the risk assessment, developers evaluate whether the development should continue.</u>
Clean Room	<u>The clean room model is used for the development of high-quality software. All levels of development are tested for bugs and defects with the goal of finding problems before they can mature. The goal of the clean room method is that the application will be bug-free at the time of release.</u>
Extreme Programming	<u>The extreme programming model values simplicity, feedback, courage, and communication. It simplifies planning to bring the entire team of developers, managers, and customers together so that adequate feedback and evaluations can be provided. This model usually works quickly, but the end product tends to be fragmented.</u>

Other concepts related to software development models include:

Concept	Description
Object-Oriented Programming	<u>Object-oriented programming, also known as OOP, is based on the organization of objects rather than actions. It uses pre-assembled programming code in a self-contained module that encapsulates a segment of data and its processing instructions. A block of programming code can be used in any number of different programs once it is written.</u> This method of development revolutionized computer program development.
Computer-Aided Software Engineering (CASE)	<u>Computer-aided software engineering is a method of using computers to help with the systematic analysis, development, design, and implementation of software.</u> It has grown to include visual programming and object-oriented programming. This model facilitates the overall security and development of applications and is best for complex, large-scale projects.

TestOut Corporation All rights reserved.