

The **Waterfall Model** is the earliest method of structured system development. Although it has come under attack in recent years for being too rigid and unrealistic when it comes to quickly meeting customer's needs, the **Waterfall Model** is still widely used. It is attributed with providing the theoretical basis for other **Process Models**, because it most closely resembles a "generic" model for software development.

Figure 2. Waterfall Model

The **Waterfall Model** consists of the following steps:

- **System Conceptualization.** System Conceptualization refers to the consideration of all aspects of the targeted business function or process, with the goals of determining how each of those aspects relates with one another, and which aspects will be incorporated into the system.
- **Systems Analysis.** This step refers to the gathering of system requirements, with the goal of determining how these requirements will be accommodated in the system. Extensive communication between the customer and the developer is essential.
- **System Design.** Once the requirements have been collected and analyzed, it is necessary to identify in detail how the system will be constructed to perform necessary tasks. More specifically, the System Design phase is focused on the data requirements (what information will be processed in the system?), the software construction (how will the application be constructed?), and the interface construction (what will the system look like? What standards will be followed?).
- **Coding.** Also known as programming, this step involves the creation of the system software. Requirements and systems specifications from the System Design step are translated into machine readable computer code.
- **Testing.** As the software is created and added to the developing system, testing is performed to ensure that it is working correctly and efficiently. Testing is generally focused on two areas: internal efficiency and external effectiveness. The goal of external effectiveness testing is to verify that the software is functioning according to system design, and that it is performing all necessary functions or sub-functions. The goal of internal testing is to make sure that the computer code is efficient, standardized, and well documented. Testing can be a labor-intensive process, due to its iterative nature.

Problems/Challenges associated with the Waterfall Model

Although the **Waterfall Model** has been used extensively over the years in the production of many quality systems, it is not without its problems. In recent years it has come under attack, due to its rigid design and inflexible procedure. Criticisms fall into the following categories:

- Real projects rarely follow the sequential flow that the model proposes.
 - At the beginning of most projects there is often a great deal of uncertainty about requirements and goals, and it is therefore difficult for customers to identify these criteria on a detailed level. The model does not accommodate this natural uncertainty very well.
 - Developing a system using the **Waterfall Model** can be a long, painstaking process that does not yield a working version of the system until late in the process.
-