# Stages in the Software Design Life Cycle

## Analysis

At the first stage, the systems analyst or programmer has to find out precisely what is required of the software. This is done by market research in the case of off-the-shelf software or interview and negotiation with a particular client. At the end of this stage, the user requirements have to be transformed into a series of bullet points or similar so that the software can be checked against this design specification at later stages of development.

## Design

The design is probably the biggest stage in the cycle as most of the development work goes on here. Firstly, the user interface has to be designed, taking into consideration what inputs and outputs the program needs and also the level of computer expertise of the user group. For example, cash dispenser software has to be usable by everyone, whereas serious gamers expect a very complicated interface.

The detail design is usually carried out using structure diagrams and/or pseudocode to refine the programmer's ideas stage by stage into smaller pieces until it is ready for coding. This is known as stepwise refinement and normally involves a top-down approach to programming.

## Implementation

At the coding stage, the Integrated Design Environment (IDE) of programming languages such as Visual Basic provide a set of tools for the rapid development of the code. Drag-and-drop user-interface design tools, text editors, syntax checkers, debugging tools, interpreters, compilers, module libraries and packaging tools are just some of the facilities which the programmer now takes for granted.

## Testing

Although the programmer tests the code as he or she codes the program, there has to be extensive testing of the finished product to expose any flaws before the software goes into normal use. The programmer plans test data to test as many options in the program as possible without spending forever repeating similar options. This is known as a test strategy and the professional programmer will test for valid data, invalid data and borderline to see if the software behaves in the expected manner but will also try to crash the program. If he can, someone else could. Usually the software is then released to a select technical group of users for further testing and reporting of faults. This is known as beta testing.

## Documentation, Evaluation and Maintenance

The finished code has to have the appropriate readme files, tutorials and user guides written. It is then evaluated to see if improvements could be made and these can be implemented as part of the maintenance programs. Bug fixes, more efficient programming and the addition of new features come into this category.