

## Key vocab

Application	Software designed to carry out a useful real-world task.
Utility	A single-purpose program normally used in the maintenance of computer systems.
Program	A self-contained set of instructions that can be stored and used by the processor.
Software	The general term for computer programs.
Programming language	A means of writing programs in a form that can be passed to a computer to process.
Instructions	A set of commands that a processor can recognise and act upon.
Open-source software	Software where the source code is made freely available, where users can legally alter the source code to create their own software. (eg. Mozilla Firefox)
Proprietary software	Software where only the compiled code is released – therefore modification, copying and redistribution of this software is legally restricted. (eg. Microsoft Office, Adobe)
Custom-written software	Software which has been custom made by software developers for an organisation
Off-the-shelf software	Generic software that provides many features that the majority of users will want
Utility system software	Software designed to help maintain or configure a computer.
Software repositories	Servers where free or open source software is available to be downloaded.
Package management software	Software used to ensure that the correct files are downloaded from a server when downloading a program.

## OPEN SOURCE vs PROPRIETARY CMS



### Open source systems

#### PROS

- + Free
- + Large developer base
- + Global third party communities for training, support and plugins
- + Multiple modules and wide input
- + Quick to set up
- + Often cleaner and better code as can be seen by others
- + Scalable
- + Portable and adaptable

#### CONS

- Support not assured
- Lack of investment
- Security fixes not guaranteed
- May not be optimal for enterprise-level websites

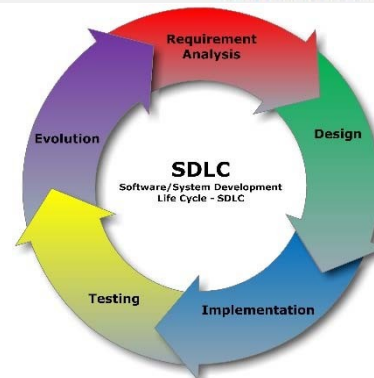
### Proprietary systems

#### PROS

- + Better documentation
- + Assured fixes to security issues
- + Better support
- + Built for enterprise-level sites
- + Training provided
- + Established third-party vendor relationships

#### CONS

- Expensive license fees
- Not a bespoke solution
- More narrow input from developers
- Lengthy set-up process
- Updating processes may be complicated
- Hard to migrate
- Specific coding knowledge required
- Limited or no APIs



## Software development stages

Analysis	Looking at a problem as a whole a decomposing into smaller parts. Success criterion should be the outcome from this stage.
Design	Creating a plan for how software will evolve. Typical design will include flowcharts, pseudocode and a testing plan. Focus on the algorithms developed and the robustness of the program.
Development	Creation of a piece of software. Clear communication about the processes, which are occurring and any solving of errors encountered, must be discussed.
Testing	The process of identifying whether software copes with expected, unexpected and boundary data and how it will respond to such situations.
Demonstration	Showing how a product works and the steps which may be taken to
Evaluation	A summary of how well a program works and how it fills the success criterion developed in the analysis stage.