**Introduction to Software Engineering**

Table of Contents

[Attributes of Good Software 3](#_Toc83392842)

[Industry Standard Software 4](#_Toc83392843)

[Software Developers and Software Engineers 5](#_Toc83392844)

[Challenges of Software Engineering 6](#_Toc83392845)

[Ethics 7](#_Toc83392846)

[Software Process 8](#_Toc83392847)

Software is a set of instructions and its associated documentation that tells a computer what to do or how to perform a task. The documentation, in particular, is not something we tend to consider. However, good documentation is extremely important for further maintenance of the code, whether by the original writer or by others.

## Attributes of Good Software

Good software has 4 attributes:

|  |  |
| --- | --- |
| **Maintainability** | Software must be written in a way so that it can be updated to meet the changing demands of the customer. |
| **Dependability and Security** | Software dependability includes a range of characteristics including reliability, security and safety. Dependable software should not cause physical or economic damage in the event of system failure. Malicious users should not be able to access or damage the system. |
| **Efficiency** | Software should not make wasteful use of system resources such as memory and processor cycles. Efficiency therefore includes responsiveness, processing time, memory utilisation, etc. |
| **Acceptability** | This means that the software must be understandable, usable and compatible with other systems that the users use. |

## Industry Standard Software

Industry standard software must meet even more requirements. They must be:

|  |  |
| --- | --- |
| **Operational** | This could be in terms of budget, efficiency, usability, dependability, correctness, functionality or security. |
| **Transitional** | This means the software must be interoperable, reusable, portable and adaptable. |
| **Easy to Maintain** | This could be in terms of flexibility, maintainability, modularity or scalability. |

Industry standard software must also face the challenges of **time**, **cost** and **quality**, since they will likely have to meet some minimum standard of quality, stay within a budget and also be finished within a specified time.

Based on all of this, software engineering can be defined as a discipline where the aim is to produce quality software, delivered on time, within budget and satisfying the requirements of the client.

## Software Developers and Software Engineers

Generally, Software Engineers tend to earn a lot more than Software Developers. Essentially, Software Developers are programmers. Their only job is to write programs. However, Software Engineers have a much wider range of responsibilities.

* They apply principles of Software Engineering to design, develop, maintain, test and evaluate computer software.
* They work with other components of the hardware system.
* They create tools to develop software, whereas Software Developers only use the tools.
* Essentially, they do the work of Software Developers on a much larger scale.

## Challenges of Software Engineering

The main challenges in the field of Software Engineering are:

* Software being used in safety-critical areas like space, aviation, nuclear power plants, etc. have a huge cost of failure since they can put lives at risk.
* There is an increased market demand for software to be built as quickly as possible while still maintain quality and staying within budget.
* The complexity of software and the need for new applications is increasing.

## Ethics

The ACM/IEEE-CS Joint Task Force on Software Engineering Ethics and Professional Practices has created a set of [ethical principles](https://ethics.acm.org/code-of-ethics/software-engineering-code/) that should be followed when working as a software engineer.

1. **PUBLIC** - Software engineers shall act consistently with the public interest.
2. **CLIENT AND EMPLOYER** - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
3. **PRODUCT** - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.
4. **JUDGMENT** - Software engineers shall maintain integrity and independence in their professional judgment.
5. **MANAGEMENT** - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
6. **PROFESSION** - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
7. **COLLEAGUES** - Software engineers shall be fair to and supportive of their colleagues.
8. **SELF** - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

## Software Process

A software process is a set of activities that leads to the production of a software product.

There are a few standard software processes, but there is no fixed process that is used throughout the industry. Every organization has their own approach to software development. For systems that are critical, a very structured development process, such as the Waterfall model, is required, whereas business systems, where requirements change frequently, would need a more flexible, agile process.

Even with the differences in software processes, there are a few activities that are common to all software processes:

1. Software Specification – The functionality of the software and the constraints on its operation must be defined.
2. Software Design and Implementation – The software that meets the specifications must be produced.
3. Software Validation – The software must be validated to ensure that it meets the specifications.
4. Software Evolution – The software must evolve to meet changing customer needs.