Chapter 1: Introduction

- Software is **computer programs** and associated **documentation**

- Software engineering is an **engineering discipline** that is concerned with all aspects of software production and -development including theories, methods and tools

- Software costs **more to maintain** than it does to develop (60% dev, 40% testing)

- Can be **Generic** (sold on market, specs set by developer) or **Customized** (by commission, specs set by customer)

-Fundamental activities include Software **specification**, software **development**, software **validation** and software **evolution**

**- Essential software product attributes are:**

**Maintainable:** Can evolve to meet the changing needs of customer

**Dependable, Reliable, & Secure:** Should not cause physical or economic damage in the event of system failure

**Efficient:** Should not make wasteful use of system resources such as memory and processor cycles

**Acceptable:** Understandable, usable and compatible with other systems that the customer uses

**-General issues that affect Software Engineering:**

**Heterogeneity:** Must be usable on different types of computer and mobile devices

**Business & social change:** Execute on a remote computer and are accessed by users (ex: web applications)

**Security & trust:** software control systems that control and manage hardware devices

**-Types of applications:**

**Stand-alone:** run on a local computer and don’t need network connection

**Interactive transaction-based:** New technology calls for rapid development of new software

**Embedded control:** Software must be trustable

**Batch processing:** Designed to process data in large batches of inputs to create corresponding outputs

**Entertainment:** Personal use and user entertainment

**Modelling and simulation:** Model physical processes or situations

**Data collection:** Collect data from sensors and send that to other systems for processing

**System of Systems:** Composed of a number of other systems

**System Engineering**

All aspects (including hardware)

**Software Engineering**

General processes & Practices

**Computer Science**

Theory and fundamentals

**-fundamental principles:**

**Managed and understood development process - Dependability and performance – Understanding what the software should do - reuse software that has already been developed rather than write new software**

**-Web:**

- Web based systems are increasingly **developed more** than local systems

**- Cloud computing** is an approach to computer services where the application runs entirely on the “cloud”

**- Software reuse** is the dominant approach for constructing web-based systems

- Web-based systems should be developed and delivered **incrementally**

- User interfaces are **constrained** by the capabilities of web browsers => technologies such as AJAX, though tough to use, allow the creation of rich interfaces

- The fundamental notions of software engineering are universally applicable to all types of system development.

- The fundamental ideas of software engineering are applicable to all types of software system.