SOFT2412

WEEK 1

**Agenda**

– Software Engineering, Software Development, SD process, SDLC, SDLC/SD models

– Agile, agile software development, software development methods (waterfall, agile, spiral)

– Agile principles, agile practices

Software **Engineering**-what and why?

What:

* Software is using everywhere : power, communication, edu, health, etc.
* Improving working efficiency and creating economy
* Emerging challenge: security, tech innovation

Why:

* Need to build **high-quality software** systems under **resource constraints**
* Social: user need, convenience
* Economical: cost
* Time: deliver of product

**Engineer vs developer:**

Engineer: design architecture, provide methods, managing development process& complexity,

Developer: write codes.

**SD process**

– Software Development Process

– A structured set of activities required to develop a software system

– Defines how various activities are to be done, and in what order

– It defines a Lifecycle for a Software Development project

– Includes processes, a set of tools, definitions of the artefacts, etc.

* universally applicable SD process:

– **Specification** – defining what the system should do;

– **Design and implementation** – defining the organization of the system and implementing the system;

– **Validation** – checking that it does what the customer wants;

– **Evolution** – changing the system in response to changing customer needs

– planning

– Plan-driven (plan-and-document heavy-weight)

– Agile processes (light-weight)

**Software Process Models(SDLC)**

– **Waterfall Model** – Development process activities as process phases

– **Spiral Model** – Incremental development risk-driven

– **Agile Model** – Iterative incremental process for rapid software development

– **The Rational Unified Process** (RUP or UP) – Bring together elements of different process models – Phases of the model in timer (dynamic perspectives), process activities (static perspective), good practices (practice perspective)

**Waterfall**: strictly defined separate phases, passing from team to team.

Pros: easy to understand and implement, used for large systems engineering projects

Cons: difficulty of accommodating change as phases too strict

**Spiral**: each loop represent a phase, successive iteration.

Pros: better iteration, risk driven approach

Cons: not practical

**Rational unified process**: Iterative and incremental(like agile)

**Agile Development Model**

Agile intend to develop systems **more quickly** with **limited time spent on analysis and design**

Why agile important:

• constant introduction of new technology

• New players enter the market,

• New requirements are added

• “Small is Beautiful”

• listening to the customer reduce failure by a smaller, more flexible competitor

• reduce maintenance costs

Agile process:

– **Light-weight**

– **People-based** rather than Plan-based