

# Software Engineering

M	T	W	S	S
Page No.:				
Date:				

(Lecture 1)

## L1 Deconstructing the Software dev. Process - Int.

- process to create softwares with comp. process
- 1. software process 2. tools (to build softwares)
- 3. code (organized & best practices)
- reflection spots

## L2 Thinking of Software in terms of Components

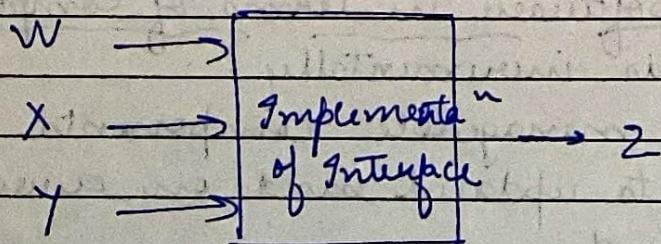
- software builds incrementally.
- divide into manageable components.
- inventory gets updated based on current purchasing & seasonal trends.
- the homepage gets customized based on my shopping & viewing history
- payment gateway → example of 1 component
- software can be divided into separately addressable components / modules that are integrated to satisfy requirements

## L3 SD Process - Requirement Specification

- 1st step of creating a software - yes, PL, etc.
- study existing components of the system, look at similar systems to understand features
- Requirements
  - goals the implemented sys. should have
  - should cater to the needs of clients
- client - ext. user, internal to your company (int. products sys.), software (mayor pay)
- think abt. who is going to use our software, for what purpose & in what way.
- need to ensure that req. of clients are met.

## L4 SD Process - Software Design & dev.

- diff. dev. may have diff. ideas about how they functionality should be implemented.
- adding new features - big picture view of the sys.
- Design
  - big picture view of the sys.
  - provides structure to the software



- Development
  - write code based on req. & design
  - usually distributed
  - developer documentation & precise interface defn.

## L5 Testing & Maintenance

- testing is done to ensure that software behaves as per the req., many bugs might exist, failure to address bugs cause catastrophes.
- Unit, Integration, Acceptance testing
  - ↳ x-test - within company
  - ↳ y-test - actual users
- Maintenance - after feature is rolled out, monitor how users are using the software.
  - monitoring
  - change the code for upgrades
  - add features

## L6 SD Models - Waterfall (Plan & Document) Model

- exp. in pers. persj. → build & fix
- SE - well-defined set of activities - SDLC
- SDLC - diff. stages over which a software evolves from the initial customer request to a fully dev. software.
- Plans & doc. perspective → waterfall model
- drawbacks - ↑ in cost, time, clients may not know what they need, confused designers, quite long (6-18 m)
- Prototype Model - build a working prototype before dev. of actual. It is discarded later.
- Spiral model - incrementally build the software
  - ↳ get feedback, refine → waterfall + prototype
  - ↳ iterations still take time.

## L7 SD - Agile

- Agile Manifesto
  1. individual interactions over processes
  2. working software over documentation
  3. customer collaboration over contract negotiation
  4. responding to change over following a plan
- incremental development
- Extreme persj. (XP), Scrum (Sprints), KanBan (Board)
- rather than practices, focus on the broad philosophy.