Week1

Week 1 SE and Software Process   
What is a *process*? Give some examples of process for domains other than IT and engineering.

The objective of software development is to anciently and

predictably deliver a software product that meets the needs of the

community of its stakeholders.

A process is a set of ordered steps intended to reach an objective.

A software development process (or model) is an approach to

building, deploying and maintaining software. The motivation

behind de\_ning a process for software development is to manage

the size and complexity of software systems.

* Efficient
* Predictable
* Successful delivery of a product as required (meets the needs). Indicate the Difference between wants and needs.

Breakdown by Activity:

Linear Process Model

1 month requirements

1 month analysis

2 month design

2 month coding & testing

Breakdown by Functionality:

Iterative Process Model

1st iteration has 20% functionality

2nd iteration adds 30%

etc.

Why should a process be *systematic*, *measurable* in terms of cost, time, and resources, and *documented*?

The process of solving customers problems by the systematic

development and evolution of large, high-quality software systems

within cost, time and other constraints.

* Systematic: This is what the discipline software Engineering is all about. We need better quality control of the software dev. Process. Bring the systematic approach from engineering to software dev. .

What are the key differences between a small software development assignment, and a software engineering project?

* Individual vs. Team
* Assignment means output is clearly stated. Project: you need to formulate/define your input and output.
* Programming vs. Process of software development
* More reviews, quality control needed.

As a developer on a SE project, who would you need to communicate with? Be as inclusive as possible.

customers other components

* Stakeholders: Project owner, Business Analyst, Users, Testers, Requirements

Why is it recommended to follow the *question - listen - paraphrase - report back - get confirmation* process when communicating?

* Explain each step
* Needs (customers, users, user stories, project describe )
* Requirement (system feedback, user cases)
* Functional (user cases, func)
* Non-functional()
* Usability, understandability, extendibility, design constraints

How are risks handled in the Unified Process?

* + RUP is Risk Focused
  + Project team to focus on addressing the most critical risks early in the project life cycle.
  + The deliverables of each iteration; especially in the Elaboration phase, must be selected in order to ensure that the greatest risks are addressed first.

Major risks \mitigated" during inception.

All risks mitigated by end of elaboration.

What are the common risks to a software project?

* Schedule (Wrong time estimation, Resources tracking: staff/systems /skills, Failure to identify complex functionalities and time required to develop those functionalities, project scope expansions)
* Budget (over-budget due to low estimate or scope expansion)
* Operational (priority conflicts, responsibilities not clear,  Insufficient resources,  No proper subject training, bad resource planning, bad communication in team)
* Technical (Requirements, lack of technology, project too complex, integration)

How do you mitigate risks?

RUP Elaboration phase implements core architecture and critical (high risk, high value) use cases. So as to drive-down the risk of the entire project

Risk driven

Major risks \mitigated" during inception.

All risks mitigated by end of elaboration