Chapter 3

Software Process Structure

Slide Set to accompany

Software Engineering: A Practitioner's Approach

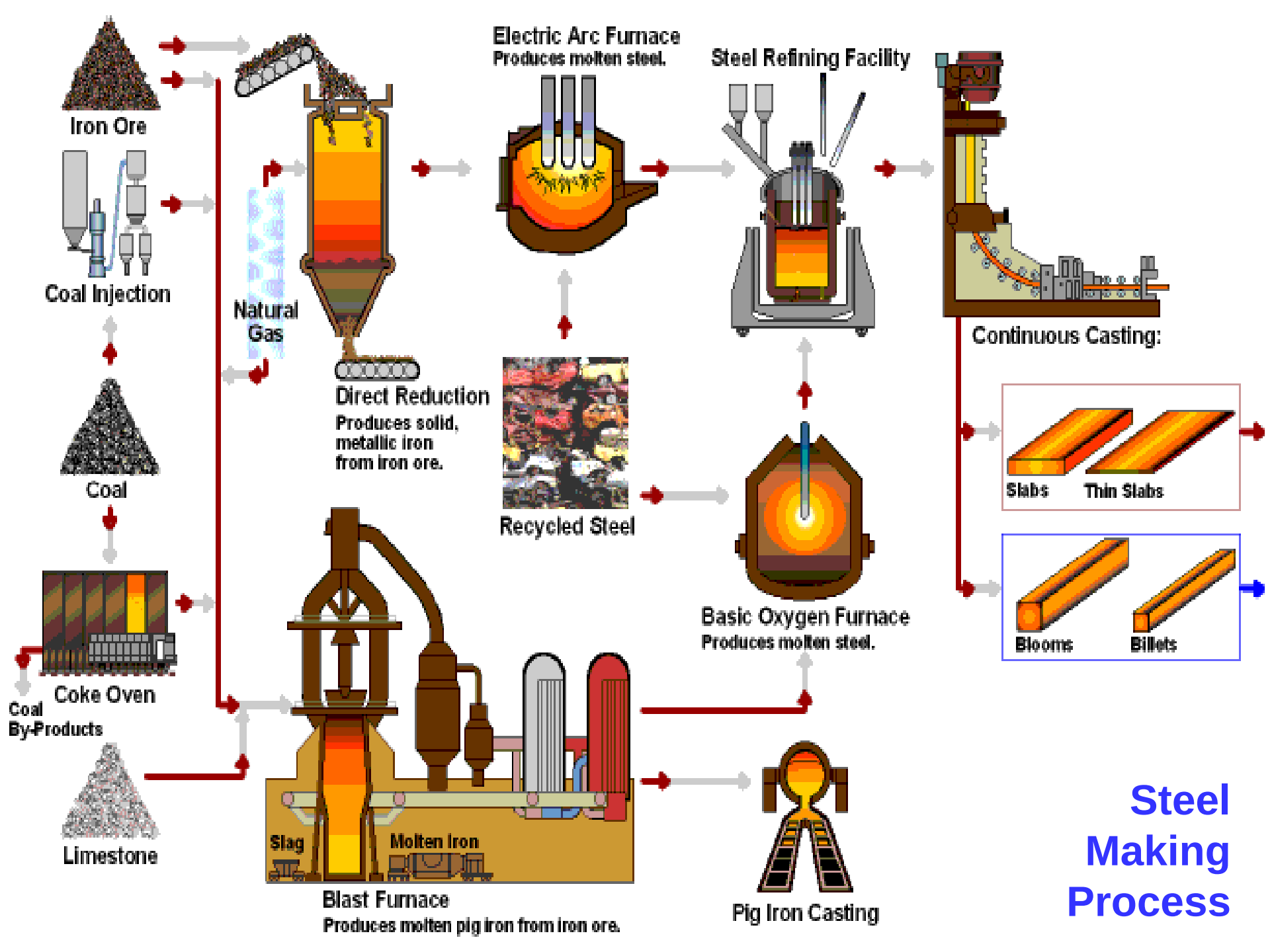
by Roger S. Pressman and Bruce R. Maxim

Table of Contents

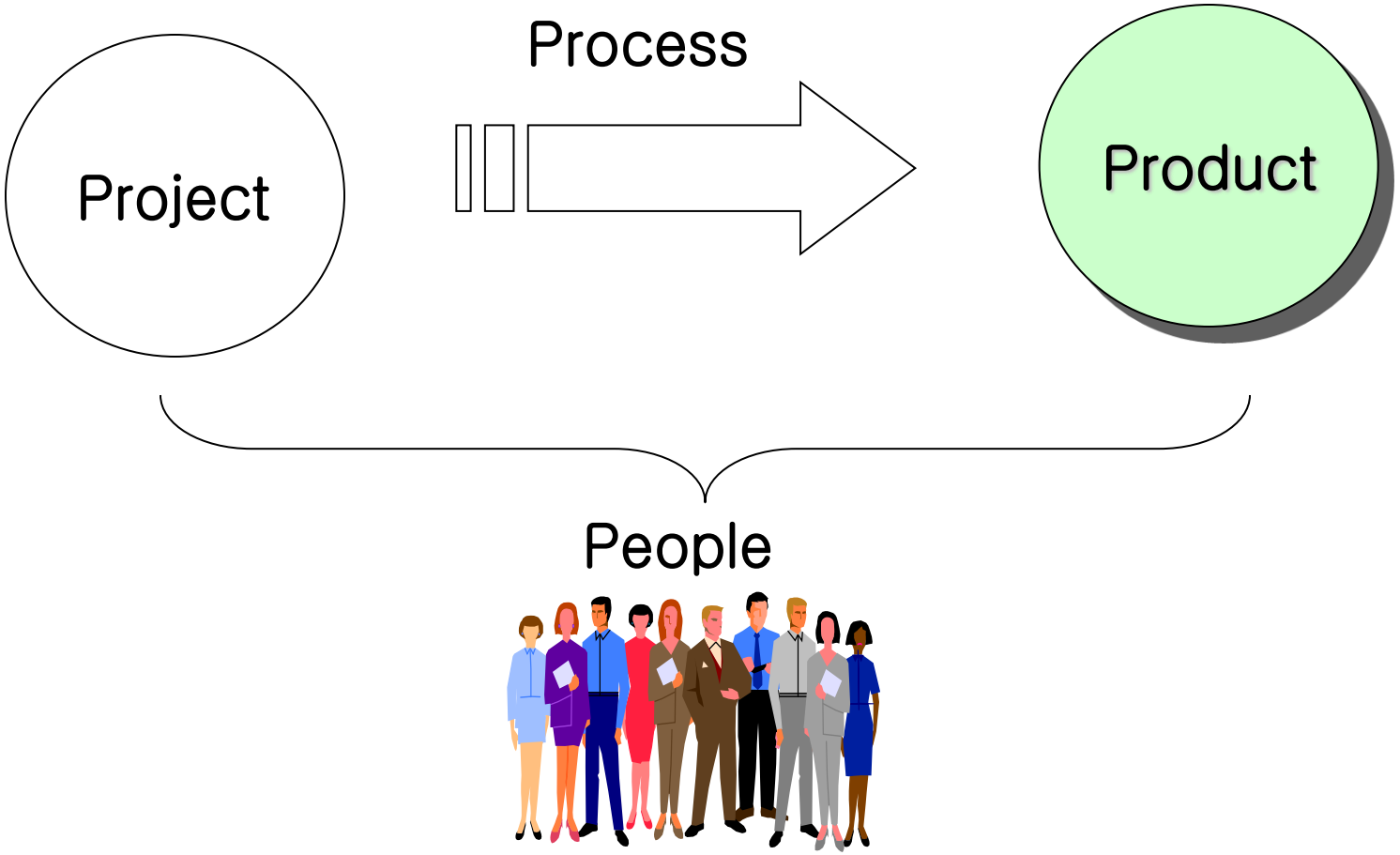
Process

"A series of actions or operations conducing to an end; *especially,* a continuous operation or treatment especially in manufacture."

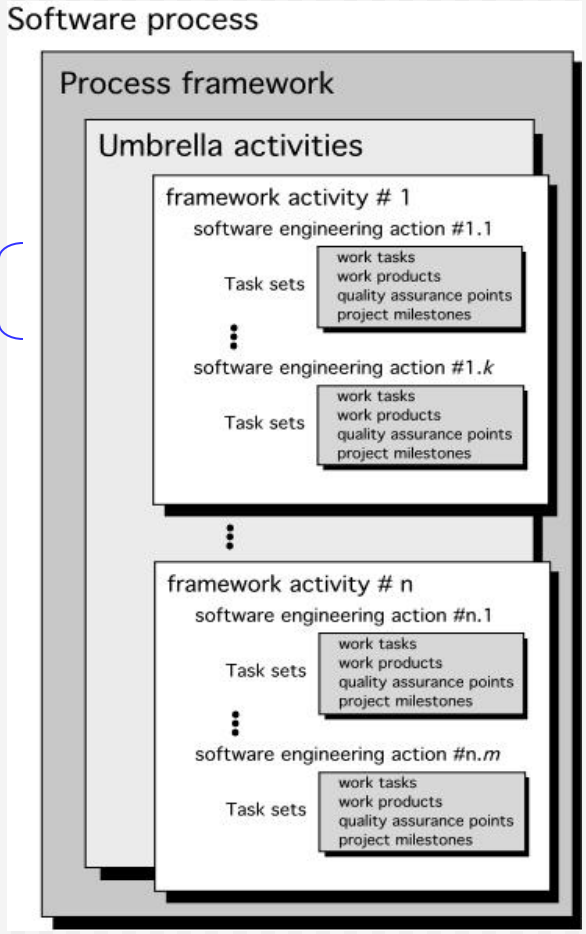
- Merriam-Webster Dictionary


Iron Ore
Coal Injection
Natural Gas
Coal
Coke Oven
Coal By-Products
Limestone
Direct Reduction
Produces solid, metallic iron from iron ore.
Electric Arc Furnace
Produces molten steel.
Recycled Steel
Steel Refining Facility
Basic Oxygen Furnace
Produces molten steel.
Continuous Casting:
Slabs
Thin Slabs
Blooms
Billets
Slag
Molten Iron
Blast Furnace
Produces molten pig iron from iron ore.
Pig Iron Casting
Steel Making Process

Four components of software development



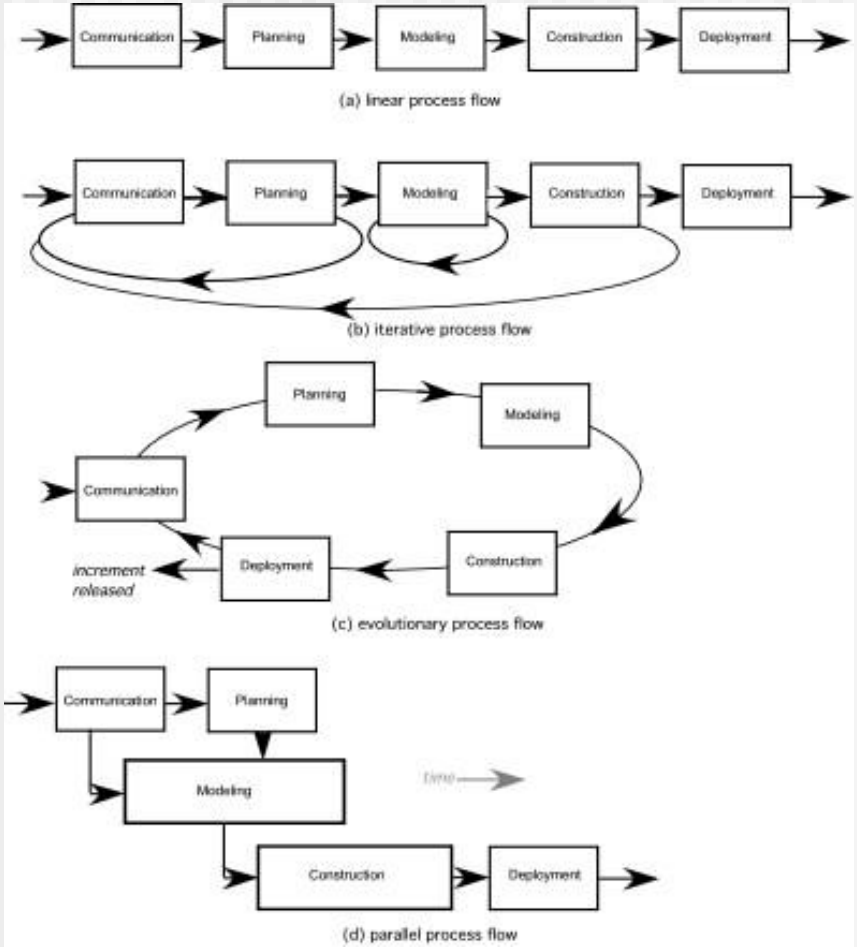
3.1 A Generic Process Model



Task set

Framework Activity

3.2 Defining a Framework Activity



3.3 Identifying a Task Set

- A task set defines the actual work to be done to accomplish the objectives of a software engineering action.
  - A list of the tasks to be accomplished
  - A list of the work products to be produced
  - A list of the quality assurance filters to be applied

3.4 Process Patterns(Skip)

- A *process pattern*
  - describes a process-related problem that is encountered during software engineering work,
  - identifies the environment in which the problem has been encountered, and
  - suggests one or more proven solutions to the problem.

- Stated in more general terms, a process pattern provides you with a *template* [Amb98]—a consistent method for describing problem solutions within the context of the software process.

Process Pattern Types(Skip)

- *Phase pattern*—defines the sequence of framework activities that occur within the process

  Example) Spiral Model, Prototyping

- *Stage pattern*—defines a problem associated with a framework activity for the process.

  Example) EstablishingCommunication

- *Task pattern*—defines a software engineering action or work task that is relevant to successful software engineering practice

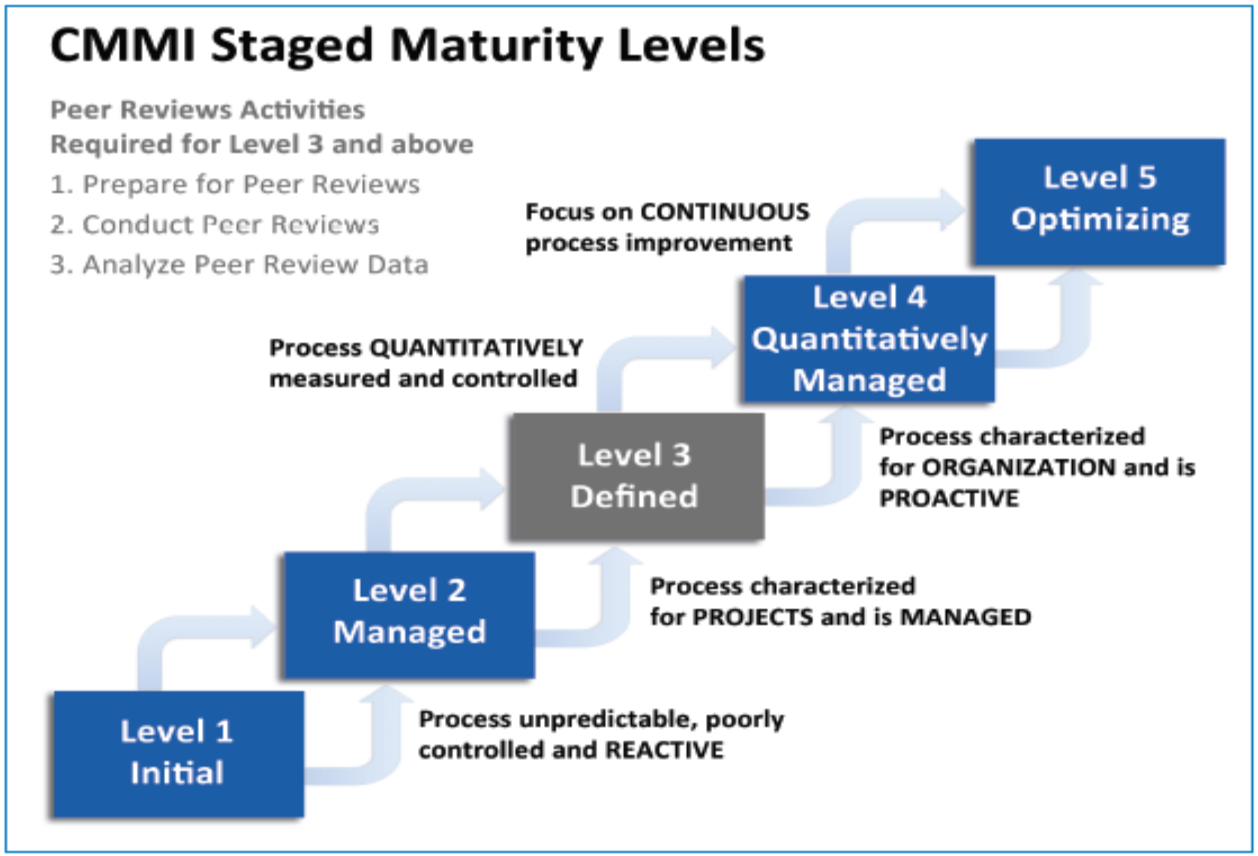
  Example) RequirementsGathering

3.5 Process Assessment and Improvement

- **Standard CMMI Assessment Method for Process Improvement (SCAMPI)** — provides a five step process assessment model that incorporates five phases: initiating, diagnosing, establishing, acting and learning.

  The Capability Maturity Model (CMM) is a measure of the effectiveness of a software process. (See the next slide.)
- **CMM-Based Appraisal for Internal Process Improvement (CBA IPI)**— provides a diagnostic technique for assessing the relative maturity of a software organization; uses the SEI CMM as the basis for the assessment [Dun01]
- **SPICE—The SPICE (ISO/IEC15504)** standard defines a set of requirements for software process assessment. The intent of the standard is to assist organizations in developing an objective evaluation of the efficacy of any defined software process. [ISO08]
- **ISO 9001:2000 for Software—**a generic standard that applies to any organization that wants to improve the overall quality of the products, systems, or services that it provides. Therefore, the standard is directly applicable to software organizations and companies. [Ant06]

CMMI - A Process Maturity Model



CMMI - Capability Maturity Model Integrated

Effective Meeting (1/4)

- Agenda
  - Review action items from last meeting
  - Check the progress of the project against the schedule
  - Discuss new issues (elicit in meeting, too)
  - Assign new action items
  - Schedule next meeting
- Minutes
  - Document
    - attendance (note latecomers)

      Send an email update if you can't attend!
    - progress on old action items
    - decisions and pending issues
    - new action items
  - Email to all members

Effective Meeting (2/4)

- Meeting Roles
  - Moderator
    - Run the agenda, keep discussion focused and concise
    - Make sure all voices are heard
    - Tangents noted for later, or saved for a sub-meeting
  - Scribe
    - Responsible for meeting minutes
  - Timekeeper
  - For your project, it is recommended that meeting roles should rotate among team members (week to week or phase to phase).

Effective Meeting (3/4)

- Keep meetings brief
  - One hour should suffice, except for a working meeting
  - The more people, the shorter the meeting (e.g. whole-project meetings should be kept short)
  - Spawn sub-group meetings to discuss substantive issues in more detail
  - Subgroups report back in the main project meeting

Effective Meeting (4/4)

- Participation is essential
  - Arrive on time, don’t leave early
  - Everyone has a voice
  - If absence is unavoidable:
    - Send an update to the meeting leader via email
    - Read meeting minutes ASAP and clarify if necessary
    - Don’t allow your absence to disrupt the project

Meeting structure for this course

Team Members			TAs		
Facilitator	Project Lead	Other Roles	Client	Mentor	Technical Advisor
✓					
Status Meeting			✓		
Q&A	✓	✓		✓	✓

Meeting Progress ↓

Mentor : observes and asks questions

Technical advisor: answer questions and provide technical advices

: typically does not exist in the real world projects