Module 2

Software Process Models

▼ What is a task set?

it defines the actual work that needs to be done to accomplish the objectives of a software engineering action

▼ What is requirements gathering (elicitation)?

- it allows stakeholders to come to a common understanding on what they want from the software that is to be built
- the level of detail is dependent on the size of the project
- it occurs during the communication phase of the project

▼ What does the task set look like for a smaller project?

1. Make a list of stakeholders for the project.
2. Invite all stakeholders to an informal meeting.
3. Ask each stakeholder to make a list of features and functions
4. Discuss requirements and build a final list.
5. Prioritize requirements.
6. Note areas of uncertainty.

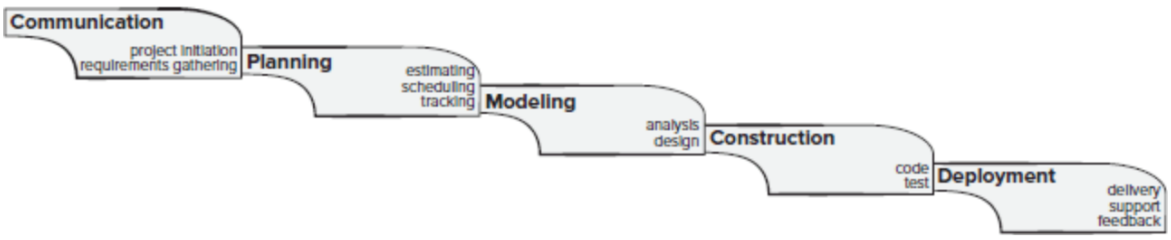
▼ What does the task set look like for a bigger project?

1. Make a list of stakeholders for the project.
2. Interview each stakeholder separately to determine overall wants and needs.

3. Build a preliminary list of functions and features based on stakeholder input.
4. Schedule a series of facilitated application specifications meetings.
5. Conduct meetings.
6. Produce informal user scenarios as part of each meeting.
7. Refine user scenarios based on stakeholder feedback.
8. Build a revised list of stakeholder requirements.
9. Use quality function deployment techniques to prioritize requirements.
10. Package requirements so that they can be delivered incrementally.
11. Note constraints and restrictions that will be placed on the system.
12. Discuss methods for validating the system.

▼ What is the waterfall model?

- the linear sequential model represents a systematic, sequential approach to software development that begins with customer specification of requirements and progresses through planning, modeling, construction, and deployment, culminating in ongoing support of the completed software
- projects do not typically follow the sequential workflow model presented in this paradigm because it is actually quite difficult for the customer to state all of the requirements in detail at the beginning of the project



▼ What is a prototype?

an incomplete version, or working model, of a product

▼ What is prototyping?

a software development model in which developers begin with a prototype which they build, test, and debug until an acceptable version is achieved

▼ What are the pros and cons of waterfall?

- Pros:
  - It is easy to understand and plan.
  - It works for well-understood small projects.
  - Analysis and testing are straightforward.
- Cons:
  - It does not accommodate change well.
  - Testing occurs late in the process.
  - Customer approval is at the end.

▼ What are the pros and cons of prototyping?

- Pros:
  - There is a reduced impact of requirement changes.
  - The customer is involved early and often
  - It works well for small projects
  - There is reduced likelihood of product rejection.
- Cons:
  - Customer involvement may cause delays.
  - There may be a temptation to “ship” a prototype.
  - Work is lost in a throwaway prototype.
  - It is hard to plan and manage.

▼ What are the pros and cons of the spiral process model?

- Pros:
  - There is continuous customer involvement.

    - Development risks are managed.
    - It is suitable for large, complex projects.
    - It works well for extensible products.
  - Cons:
    - Risk analysis failures can doom the project.
    - The project may be hard to manage.
    - It requires an expert development team.

▼ What are the pros and cons of the unified process model?

  - Pros:
    - Quality documentation is emphasized.
    - There is continuous customer involvement.
    - It accommodates requirement changes.
    - It works well for maintenance projects.
  - Cons:
    - Use cases are not always precise.
    - It has tricky software increment integration.
    - Overlapping phases can cause problems.
    - It requires an expert development team.