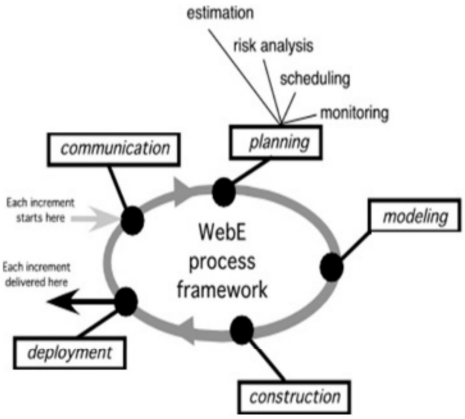
Lecture 5 Notes

Planning

Planning



- **Planning** is a key activity. But the scope of planning activities varies among people involved in a WebE project.
  - **A team leader** plans**,** monitors, and coordinates the combined work of a WebE team.
  - **A Web engineer manages** day-to-day work—planning, monitoring, and controlling technical tasks.
- Take an agile approach to planning
  - Adapt effort and time spent on planning to the complexity of the WebApp increment

Planning guidelines

- Understand scope *before* you define work tasks or schedule for an increment
- Refine framework actions and tasks
- Be sure you have the right team
- Evaluate risks
- Define a schedule
- Identify quality filters
- Identify how you'll manage change

WebApp Project Scope

- To plan effectively, you need to understand project scope
- **To establish scope be sure you understand:**
  - **Context:** How does WebApp fit into a business context, and what constraints are imposed as a result of the context?
  - **Information objectives:** What customer-visible content objects are used by the WebApp increment?
  - **Functionality:** What functions are initiated by the end user or invoked internally by the WebApp to meet the requirements defined in usage scenarios?
  - **Constraints and performance:**

- What technical and environmental constraints are relevant?
- What special performance issues (including security and privacy issues) will require design and construction effort?

Refining Actions and Tasks

- WebE framework actions and tasks are refined by "melding" characteristics of the increment and the process.
- One way to accomplish this: create a table similar to the one illustrated in the following figure.
  - **The top row** of the table lists the key content objects and functions to be delivered as part of the increment deployment.
    - **Content objects** (e.g., Walls, Doorways, and Windows) are listed in the first section of the top row.
    - **Functions** (e.g., *Specify and draw walls*) are listed next.
  - **The first column** lists the set of framework actions and tasks for modeling, construction, and deployment.

Content and functions / Framework actions and tasks	•••	Walls	Doorways	Windows	Specify and draw walls	Specify and draw doorways	Specify and draw windows	Compute size of each room	Save/retrieve a named space	Update/delete a named space	Print a named space	•••
Modeling												
Analysis												
Review user scenarios												
Show content relationships												
Create interaction model												
Elaborate content detail												
Define database requirements												
Refine function requirements												
Refine interface requirements												
Design												
Perform interface design												
Special interaction mechanics												
Refine page layout												
Show navigation mechanisms												
Perform aesthetic design												
⋮												

The Team

- *Interestingly, people working together with good communication and interaction can operate at noticeably higher levels than when they use their individual talents. We see this time and again in brainstorming and joint problem-solving sessions. Therefore, agile project teams [WebE teams] focus on increasing both individual competencies and collaboration levels.* Cockburn and Highsmith
- But how do successful teams conduct their business?
  - A set of team guidelines should be established.
  - Strong leadership is a must.
  - Respect for individual talents is critical.
  - Every member of the team should commit.
  - It's easy to get started, but it's very hard to sustain momentum.

Risk Management

- Many problems can arise during a project
- Risk management focuses on understanding and managing these problems
  - Identify the risk; Assess its probability of occurrence; Estimate its impact; Establish a contingency plan
- Considers risk at two different levels of granularity

  (1) the impact of risk on the entire WebApp project, and

  (2) the impact of risk on the current WebApp increment
- Typical risks:
  - Is the time timeframe defined and reasonable?
  - Will the increments provide ongoing value for end users
  - How will requests for change impact delivery schedules?
  - Is the available technology appropriate for the job?
  - Does the team have the right mix of skills to build this increment

➢ Identifying Risks

- Address the fundamental question: "What can go wrong?"
- Each team member is asked to make a list of risks
  - *People risks*: potential problems that can be directly traced to some human action or failing.
  - *Product risks*: potential problems associated with WebApp content, functions, constraints, or performance.

- *Process risks*: problems that are tied to the framework actions and tasks that have been chosen by the team

➢ **Risk Analysis**

1. **The WebE team performs a quick evaluation in two ways:**
   - **Probability**: the likelihood or probability that the risk will become a reality. (normally represented as a percentage),
   - **Impact:** the consequences of the problems associated with the risk, should it occur. The impact on schedule and/or cost (often represented using an ordinal scale of 1 [low] to 4 [high]).
2. Once probability and impact have been estimated, the team build **a risk table** as illustrated in the following figure:

Risks	Probability	Impact
People		
Little XML experience on team	80%	3
Stakeholders uncooperative	60%	2
Senior manager may change midstream	40%	1
Product		
Informational content may be outdated	50%	2
Algorithms may not be adequately defined	80%	3
Security for WebApp more difficult than expected	80%	3
Database integration more difficult than expected	40%	3
Space def. capability more difficult than expected	70%	3
Process		
Not enough emphasis on communication	60%	2
Too many analysis tasks (too much time spent)	30%	1
Not enough emphasis on navigation design	40%	2
⋮	⋮	⋮

3. **The risk table is sorted by:**
   - Probability and then by impact.
   - *Alternatively*, a composite score derived from *probability* x *impact* can be calculated and the table sorted on this basis.

4. High-impact risks percolate to the top of the table, and low-probability risks drop to the bottom. The team defines a ***cutoff line*** (drawn horizontally at some point in the table and implies that only risks that lie above the line will be given further consideration).

➢ **Risk Contingency Planning**

- Development time spans are short, so contingency plans are usually not formally documented.
  - Document as information notes by the team leader
- Consider each risk that falls above the cutoff line in the risk table and answer three questions:

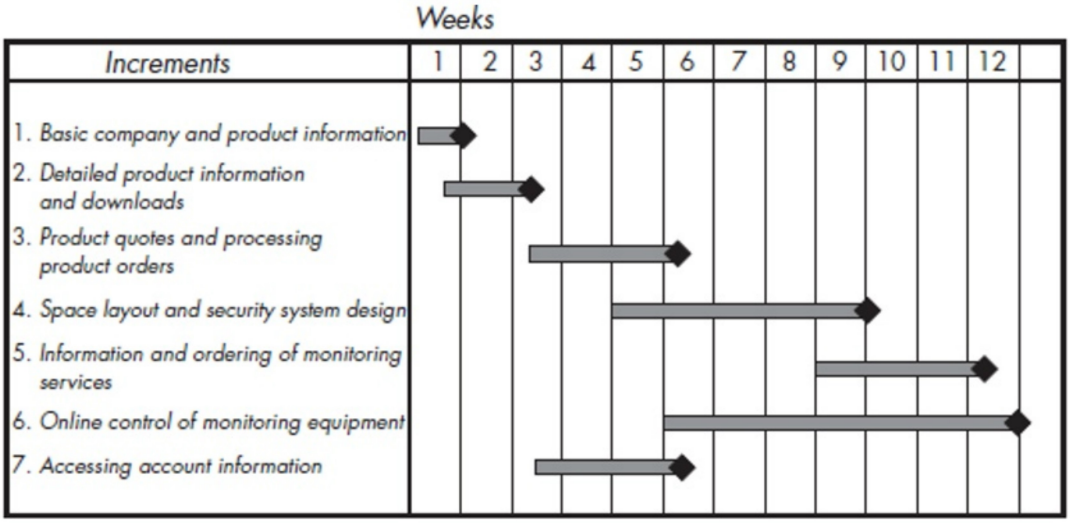
  1. How can we avoid the risk altogether?
  2. What factors can we monitor to determine whether the risk is becoming more or less likely?
  3. Should the risk become a reality, what are we going to do about it?

Developing a Schedule

- How do projects fall behind schedule? *One day at a time*, Fred Brooks
- **Approach:**
  - List all WebE actions and tasks for an increment
  - Build a network that depicts interdependencies
  - Identify tasks that are criticalT
  - Track progress (especially critical tasks)
- The WebApp schedule evolves over time.
- During the first iteration a macroscopic schedule is developed.
  - Identify all increments and dates on which each will be deployed.
- For each subsequent increment
  - The entry for the increment on the macroscopic schedule is refined into a detailed schedule.

⇨ **Note:** A **microscopic schedule** details the finest-grain activities and their timing, while a **macroscopic schedule** provides a high-level overview of major tasks and phases. In practice, this often means a macroscopic, or "big-picture," schedule is developed first, followed by a more detailed microscopic schedule that breaks down the major tasks into smaller, individual activities.

- The WebE team consults and negotiates with stakeholders and develops a preliminary deployment schedule for all increments. An example of a timeline for this schedule is illustrated in the following figure:


Weeks
Increments
1
2
3
4
5
6
7
8
9
10
11
12
1. Basic company and product information
2. Detailed product information and downloads
3. Product quotes and processing product orders
4. Space layout and security system design
5. Information and ordering of monitoring services
6. Online control of monitoring equipment
7. Accessing account information