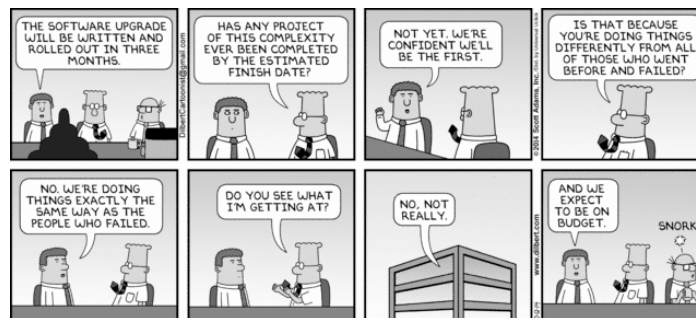


---

---

## CIS 422/522

### Project Planning



---

---

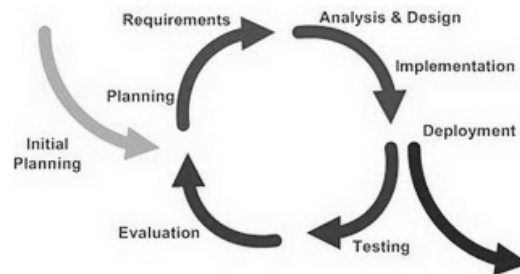
## Review: Need to Organize the Work

- Nature of a software project
  - Software development produces a set of interlocking, interdependent work products
    - E.g. Requirements -> Design -> Code -> Test
  - Implies dependencies between tasks
  - Implies dependencies between people
- Must organize the work such that:
  - Every task gets done
  - Tasks get done in the right order
  - Tasks are done by the right people
  - The product has the desired qualities
  - The end product is produced on time

## Use Iterative Process Model

---

- Process viewed as a sequence of iterations
- Addresses key risks
  - Have something to deliver
  - Feedback loop built in
- Each team will implement the abstract model differently



CIS 422/522 © S. Faulk

3

## From Process to Plan

---

- Process manifests itself in the project plan
  - Process definition is an abstraction
  - Many possible ways of implementing the same process
- *Project plan makes process concrete*, it assigns
  - People to roles
  - Artifacts to deliverables and milestones
  - Activities to tasks over time
- *Project plan is itself a product of the process*
  - Activity: project planning
  - Artifact: the Project Plan
  - Roles: Project Manager (owner), team members
- Evolves as the project proceeds

CIS 422/522 © S. Faulk

4

## Project Plan

---

- Purpose: specifies how project resources will be organized to:
  - Create each deliverable
  - Meet quality goals
  - Address developmental goals (e.g., mitigate risk)
- Audience: answers specific kinds of questions for specific types of users, e.g.:
  - General stakeholders: What is the development approach? How does it address project risks?
  - Customers: When will the product be delivered?
  - Managers: When will tasks be completed? What is the current progress against the plan?
  - Developers: What should I be working on and when?

## Plan Outline

---

- Plan contents (template in Assembla workspace)
  - Purpose and audience (who will use the document?)
  - Project background (from requirements)
  - Team roles and responsibilities
  - Risks and risk mitigation
    - What are the key risks? (Team should brainstorm this)
    - Which mitigation strategies will the project deploy
  - Process: development process, how its tailored, rationale
  - Mechanisms, methods, and techniques
    - What kinds of methods and tools will be used?
    - E.g., planning tools, design methods, IDEs, etc.
  - Detailed schedule and milestones
  - Resources and references

## Your Project Plans

---

- ***This is not an abstract, hypothetical exercise!***
- Your projects have real
  - Resources (people, time)
  - Risks (schedule, quality, etc.)
  - Process, schedule, etc.
- These must be reflected in your meetings, plans, schedules, and other work products
- This is how you demonstrate mastery of class concepts

## Detailed Schedule and Milestones

---

- Maps people to tasks over time such that
  - Personnel are fully engaged (time is not wasted)
  - Delivery meets schedule
- Answers: “Who is working on which tasks, what is their progress, and when will they be finished?”
- Inputs
  - Set of artifacts to be created (superset of deliverables)
  - Dependencies/precedence between tasks
  - People filling roles that perform tasks
  - Time budget for each task
- Output
  - Current project schedule
  - Deadline for each task
  - Sequencing among tasks
  - Allocation of people to tasks

## Project Plan Template

---

- Use the template provided in your Assembla team workspace
- This should be a *living document*
  - Changed as the project progresses
  - Ideally, always gives a current view of the *progress against the plan*
    - Shows planned activities
    - Gives snapshot of the current project state
    - This is what I am looking for (or any manager)

---

## Project Planning Tools

Work Breakdown Structure (WBS)

PERT Chart

Gantt Chart

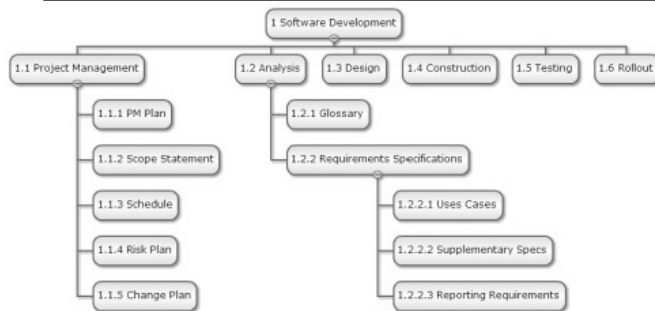
## Work Breakdown Structure

---

- Structured technique for decomposing work into individual tasks with the goals:
  - Identify the complete set of tasks in the project
  - Provide units of work (for individuals or teams)
  - Provide units of work for scheduling and costing
- Identify hierarchy of tasks and subtasks
  - Identify major tasks in project
  - Decomposing each element into component parts
  - Continuing to decompose until manageable work packages can be mapped to roles
- Works best when:
  - Tasks correspond to key deliverables
  - Sum of tasks is 100% of the work
  - Tasks do not overlap
  - Each leaf task takes about the same amount of time

## Work Breakdown Structure

---



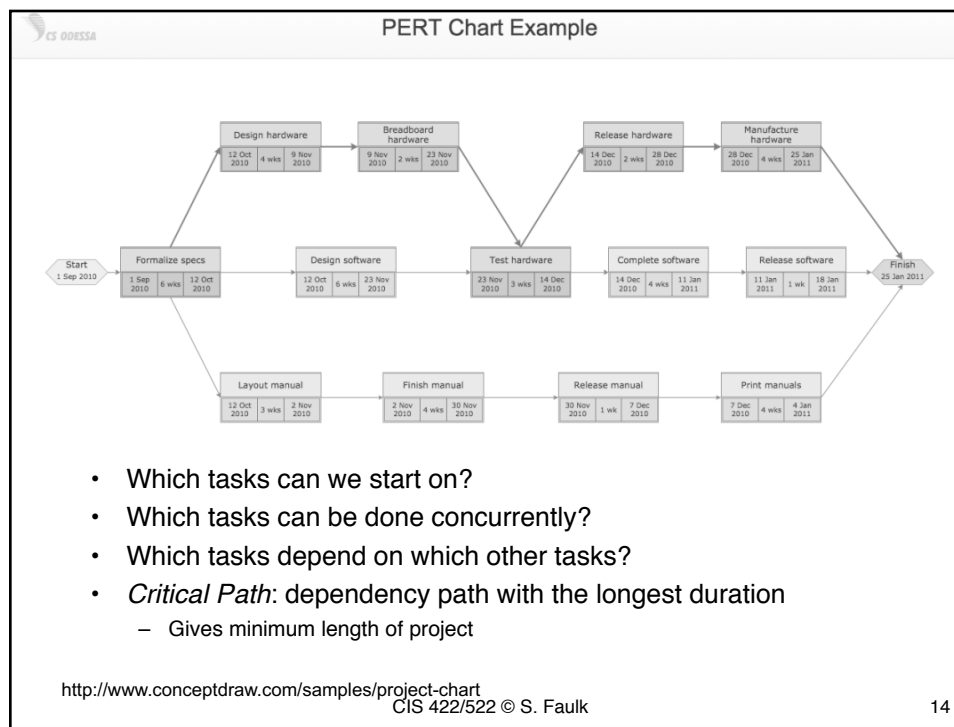
- |  |                          |
|--|--------------------------|
| <ol style="list-style-type: none"> <li>1. Software Development                     <ol style="list-style-type: none"> <li>1. Project Management</li> <li>2. Analysis                             <ol style="list-style-type: none"> <li>1. Glossary</li> <li>2. Requirements Specification                                     <ol style="list-style-type: none"> <li>1. Use Cases</li> <li>2. Supplementary Specs...</li> </ol> </li> </ol> </li> </ol> </li> </ol> | } Equivalent list format |
|--|--------------------------|

## Pert Chart

- Network analysis or PERT is used to identify dependencies between the tasks in the work breakdown structure
- Helps identify where ordering of tasks may cause problems because of precedence or resource constraints
  - Where task B cannot begin before task A ends
  - Where one person cannot do two tasks at the same time
  - Where adding a person can allow tasks to be done in parallel, shortening the project

CIS 422/522 © S. Faulk

13

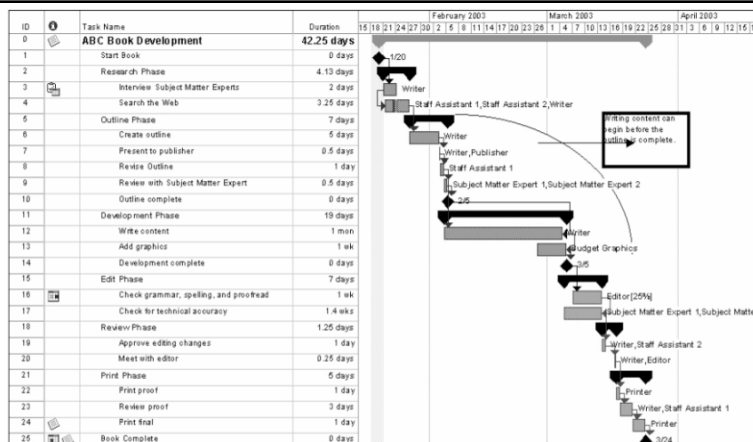


14

## Gantt Charts

- Method for *visualizing a project schedule* in one chart showing
  - The set of tasks
  - Start and completion times
  - Task dependencies
  - Responsibilities
- PERT charts can be reformatted as Gantt charts
- Typically requires a tool, e.g., <http://www.ganttproject.biz/>, smartchart

## Example Gantt Chart





## Project Milestone Planning

---

- Milestone planning is used to show the major steps that are needed to reach the goal on time
- Milestones typically mark completion of key deliverables or establishment of baselines
  - *Baseline*: when a work product is put under configuration management and all changes are controlled
- Often associated with management review points
  - E.g., Requirements baseline, project plan complete, code ready to test
- Can use Gantt or PERT charts to show milestones
- Begin with project events in Schedule

## A Simple Alternative

---

### Week 1:

Date Assigned	Due Date	Task	Person Responsible	Status	Date Completed
2/3	2/5	Brainstorm project ideas	Everyone	Complete	2/5
2/3	2/4	Set up meeting w/ instructor	Heidi	Complete	2/3
2/3	2/6	Decide on project	Everyone	Complete	2/6
2/6	2/10	Create Git repository	Heidi	Complete	2/6

### Week 2:

Date Assigned	Due Date	Task	Person Responsible	Status	Date Completed
2/10	2/10	Decide on software requirements	Everyone	Complete	2/10
2/10	2/15	Plan and design 1st iteration	Everyone	Complete	2/13
2/10	2/10	Set up meeting w/ Kathleen Freeman-Hennessy	Heidi	Complete	2/10
2/13	2/15	Write ConOps	Nicole, Heidi	Complete	3/2
2/13	2/19	Write project plan	Nicole, Heidi	Complete	2/19
2/13	2/22	Write software requirements	Nicole, Heidi	Completed	3/2
2/15	2/24	Implement 1st iteration	Dex, Hans, Yakun	Complete	2/24

## How much planning?

---

- Planning itself consumes resources; how much planning is enough?
- Enough that:
  - Everyone knows what they should be doing
  - Everyone knows what other people are supposed to be doing
  - Everyone knows when specific deliverables should be finished
    - Can track dependencies between their tasks and others
    - Know when task inputs will be available
  - It is easy to determine the current status of the project against plan
- In practice, detail decreases with distance

## Summary

---

- Project plan makes process concrete
  - People to roles
  - Artifacts to deliverables and milestones
  - Activities to tasks over time
- Plan is key to organizing the work but expect it to change
  - *The plan is nothing, the planning is everything* – D. Eisenhower
- Should understand the use of common planning tools (WBS, Pert, Gantt)

---

Questions?