5. Project planning and management

Role of a manager

Charts and Critical Path Analysis

Estimation Techniques

Monitoring
Role of a manager

Directs resources for the achievement of goals

LEADER also provides
  vision
  inspiration
  rises above the usual

No one right way to manage
Management Continuum

authoritarian
autocratic

consultative

democratic
participative

solves problems alone
dictates decisions
discusses problems
makes decision
chairperson
agrees problem
creates consensus
Managerial Roles

Henry Mintzberg (1939)

Interpersonal
- Figurehead, leader, lisaison

Informational Roles
- Monitor, disseminator, spokesperson

Decisional Roles
- entrepreneur, resource allocator, disturbance allocator, negotiator

Managerial and Leadership Qualities

Technical / Professional knowledge
Organisational know-how
Ability to grasp situations
Ability to make decisions
Ability to manage change
Creative
Mental flexibility
Learns from experience
Pro-active
Moral courage
Resilience
Social Skills
Self Knowledge
Project Management Variable

Resource

Time

Function

You can have any two of quick, good or cheap, but not all three.
Development cycle:

- Specification
- Analysis
- Build
- Test
- Maintain

Effort

Time

Alpha Beta
Crossing the Chasm

- Geoffrey Moore, after Everett Rogers
Approaches and methodologies

Top Down
- waterfall decomposition

Bottom Up
- meta machine

Rapid Prototype
- successive refinement
- agile engineering

Muddle through
In February 2001, 17 software developers met at the Snowbird resort in Utah to discuss lightweight development methods. They published the Manifesto for Agile Software Development, in which they said,

**Manifesto for Agile Software Development**

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

**Individuals and interactions** over processes and tools
**Working software** over comprehensive documentation
**Customer collaboration** over contract negotiation
**Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck  
Mike Beedle  
Arie van Bennekum  
Alistair Cockburn  
Ward Cunningham  
Martin Fowler

James Grenning  
Jim Highsmith  
Andrew Hunt  
Ron Jeffries  
Jon Kern  
Brian Marick

Robert C. Martin  
Steve Mellor  
Ken Schwaber  
Jeff Sutherland  
Dave Thomas
Agile explosion

https://en.wikipedia.org/wiki/Agile_software_development

Popular agile software development frameworks include:

- Adaptive software development (ASD)
- Agile modeling
- Agile Unified Process (AUP)
- Crystal Clear methods
- Disciplined agile delivery
- Dynamic systems development method (DSDM)
- Extreme programming (XP)
- Feature-driven development (FDD)
- Lean software development
- Kanban
- Scrum
- Scrumban

- Acceptance test-driven development (ATDD)
- Agile modeling
- Backlogs (Product and Sprint)
- Behavior-driven development (BDD)
- Business analyst designer method (BADM)[37]
- Cross-functional team
- Continuous integration (CI)
- Domain-driven design (DDD)
- Information radiators (scrum board, task board, visual management board, burndown chart)
- Iterative and incremental development (IID)
- Pair programming
- Planning poker
- Refactoring
- Scrum events (sprint planning, daily scrum, sprint review and retrospective)
- Test-driven development (TDD)
- Agile testing
- Timeboxing
- User story
- Story-driven modeling
- Retrospective
- Velocity tracking
- User Story Mapping

The Agile Alliance has provided a comprehensive online guide to applying agile these and other practices.

https://www.agilealliance.org
Scrum, Sprints, Timeboxes

Product Backlog → Sprint Backlog → Sprint → Working Increment of the software

- 30 days
- 24 h
Spiral Methodology

1. Determine objectives
2. Identify and resolve risks
3. Development and Test
4. Plan the next iteration

## Microsoft Solutions Framework 4.0

(adapted from [http://slideplayer.com/slide/6868969/](http://slideplayer.com/slide/6868969/))

<table>
<thead>
<tr>
<th>Deliverables</th>
<th>Goals</th>
</tr>
</thead>
</table>
| **Envision**                                                                 | - Vision / scope document  
  - Project structure document  
  - Initial risk assessment document  
  Develop a clear understanding of what is needed within context of project constraints  
  Assemble necessary team to envision solution with options and approaches to meet needs given constraints |
| **Plan**                                                                     | - Functional specifications  
  - Master project plan  
  - Master project schedule  
  Evolve conceptual solution into tangible designs and plans so it can be built in the build phase |
| **Build**                                                                   | - Completed solution  
  - Training materials  
  - Documentation  
  - Marketing materials  
  - Updated master plan, schedule and risk document  
  Build various aspects of the solution in accordance with plan track deliverables |
| **Test**                                                                    | - Proactive - leads build effort  
  - Supportive - follows build effort  
  Expose issues, uncover design flaws and identify unexpected behaviour |
| **Stabilise**                                                                | - Pilot review  
  - Release-ready versions of solutions and accompanying collateral  
  - Testing and bug reports  
  - Project documents  
  Improve solution quality to meet release criteria for deployment to production  
  Validate solution meets stakeholder needs  
  Validate solution usability |
| **Deploy**                                                                  | - Operations and support information systems  
  - Revised processes and procedures  
  - Repository of all solution collateral  
  Place solution into production at designated environments  
  Facilitate smooth transfer of solution from project team to operations team as soon as possible |
Scrum Meetings

Daily Scrum

Scrum of scrums

Sprint Planning Meetings

Sprint Review Meetings

Sprint Retrospective
Pertt and Gantt Charts

Visual representation of project

Microsoft Project


Example: Getting up in the morning

<table>
<thead>
<tr>
<th>Task</th>
<th>Duration (mins)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Alarm rings</td>
<td>0</td>
</tr>
<tr>
<td>2. Wake Up</td>
<td>3</td>
</tr>
<tr>
<td>3. Get out of bed</td>
<td>5</td>
</tr>
<tr>
<td>4. Wash</td>
<td>5</td>
</tr>
<tr>
<td>5. Get dressed</td>
<td>5</td>
</tr>
<tr>
<td>6. Put kettle on</td>
<td>2</td>
</tr>
<tr>
<td>7 Wait for kettle to boil</td>
<td>5</td>
</tr>
<tr>
<td>8 Put toast on</td>
<td>2</td>
</tr>
<tr>
<td>9 Wait for Toast</td>
<td>3</td>
</tr>
<tr>
<td>10 Make coffee</td>
<td>3</td>
</tr>
<tr>
<td>11 Butter Toast</td>
<td>2</td>
</tr>
<tr>
<td>12 Eat Breakfast</td>
<td>10</td>
</tr>
<tr>
<td>13 Leave for Lectures</td>
<td>0</td>
</tr>
</tbody>
</table>
Pert Chart
Critical Path Analysis

Compute earliest and latest start / finish for each task

The difference is the slack

The Critical Path joins the tasks for which there is no slack

Any delay in tasks on the Critical Path affects the whole project
<table>
<thead>
<tr>
<th>ID</th>
<th>Task</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Get out of bed</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Wash</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Get dressed</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>Put Kettle on</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>Wait for kettle to boil</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Make Coffee</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>Put toast on</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>Wait for Toast</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>Butter Toast</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>Eat Breakfast</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>Leave for Lectures</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>9am</td>
</tr>
</tbody>
</table>
Example
<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Start</th>
<th>1/5/95</th>
<th>23/6/95</th>
<th>1/5/95</th>
<th>14/7/95</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Phase 2</td>
<td>Start</td>
<td>19/6/95</td>
<td>25/8/95</td>
<td>19/6/95</td>
<td>25/8/95</td>
</tr>
<tr>
<td></td>
<td></td>
<td>22/6/95</td>
<td>26/6/95</td>
<td>17/7/95</td>
<td>28/7/95</td>
</tr>
<tr>
<td></td>
<td></td>
<td>17/8/95</td>
<td>21/8/95</td>
<td>17/8/95</td>
<td>22/8/95</td>
</tr>
<tr>
<td></td>
<td></td>
<td>22/8/95</td>
<td>26/8/95</td>
<td>17/8/95</td>
<td>22/8/95</td>
</tr>
<tr>
<td></td>
<td></td>
<td>17/8/95</td>
<td>21/8/95</td>
<td>17/8/95</td>
<td>22/8/95</td>
</tr>
</tbody>
</table>

Example Pert
<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Duration</th>
<th>Scheduled Start</th>
<th>Month 3</th>
<th>Month 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Start</td>
<td>6m</td>
<td>3/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Phase 1</td>
<td>6m</td>
<td>1/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Analyze</td>
<td>4m</td>
<td>2/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Code</td>
<td>3m</td>
<td>2/20/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Test</td>
<td>1w</td>
<td>3/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Phase 2</td>
<td>7d</td>
<td>1/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Analyze</td>
<td>2d</td>
<td>2/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Code</td>
<td>1d</td>
<td>2/20/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Test</td>
<td>2d</td>
<td>3/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Phase 3</td>
<td>16w</td>
<td>1/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Analyze</td>
<td>4w</td>
<td>2/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Code</td>
<td>2d</td>
<td>3/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Test</td>
<td>4w</td>
<td>3/19/95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The diagram shows a Gantt chart and a workload chart. The Gantt chart indicates the progress of various tasks over time, with bars representing the duration and start date of each task. The workload chart on the right side shows the allocation of resources (programmers) over a period of time, with shaded areas indicating the assigned workload for each role.
Levelling

Adjust tasks to match resources available
Automatic system available, but does not always give an optimum result
Tasks may be delayed within slack without affecting project dates
Otherwise consider extending project, or using more resource
Adding resource to a late project may cause RECURSIVE COLLAPSE
  consider carefully whether the benefits outweigh the additional learning delays and overheads
Derive costings
Larger example
Estimation Techniques

Experience

Comparison with similar tasks

- 20 lines of code / day
- can vary by 2 orders of magnitude

Decomposition

Plan to throw one away

20 working days per month BUT 200 per year
Rules of Thumb

Software projects
estimate 10 x cost and 3 x time

1/3/10 rule
1 cost of prototype
3 cost of creating a product
10 cost of sales and marketing

Hartree’s Law
The time to completion of any project, as estimated by the project leader, is a constant (Hartree’s constant) regardless of the state of the project
A project is 90% complete 90% of the time

80% rule
Don’t plan to use more than 80% of available resource
Cynic’s Project Stages

Enthusiasm
Disillusionment
Panic
Persecution of the innocent
Praise of the bystander