

Criteria for Project Evaluation (Time)

Saeed Siddik

Assistant Professor

IIT University of Dhaka

Time & Scheduling

- Time and scheduling are critical for meeting deadlines
- optimizing resource use in software projects
- Scheduling can be static (planned once at the start)
- or dynamic (adapted to changes during the project)

Software Project Scheduling Problem (SPSP)

- Definition: assigning tasks to resources under constraints.
- Assigning tasks \rightarrow resources \rightarrow timeline
- Must satisfy constraints (skills, precedence, availability)
- Objectives:
 - Minimize project duration
 - Avoid resource overload
 - Meet deadlines

Software Project Scheduling Problem (SPSP)

- Definition: assigning tasks to resources under constraints.
- SPSP as an NP-hard optimization problem.
- Search-Based Software Engineering (SBSE) perspective → use of evolutionary algorithms, metaheuristics.
- Common objectives: minimize project duration, balance workload, avoid resource conflicts.

Popular Scheduling Techniques

- Work Breakdown Structure (WBS): for task breakdown, leveling, and resource allocation.
- Critical Path Method (CPM): dependency-focused optimization
- Program Evaluation and Review Technique (PERT): for probabilistic time estimates and identifying critical paths
- Gantt Charts & Milestones: for visualizing timelines.

SPSP example: WBS → CPM → PERT

- **T1: Requirement Analysis** — duration 3 days
- **T2: Coding** — duration 6 days (depends on T1)
- **T3: Testing** — duration 4 days (depends on T2)
- PERT time estimates for each task:

Task	Optimistic (O)	Most Likely (M)	Pessimistic (P)
T1	2	3	5
T2	4	6	10
T3	3	4	7

WBS Breakdown (task module)

- Three main tasks:
 - **T1: Requirement Analysis**
 - **T2: Coding** (depends on T1)
 - **T3: Testing** (depends on T2)
- Durations (days):
 - T1 = 3 days
 - T2 = 6 days
 - T3 = 4 days

CPM Scheduling (Forward + Backward Pass)

- **Forward Pass**

- **T1:** $ES = 0 \rightarrow EF = 3$
- **T2:** $ES = EF(T1) = 3 \rightarrow EF = 3 + 6 = 9$
- **T3:** $ES = EF(T2) = 9 \rightarrow EF = 9 + 4 = 13$
- **Project Completion Time = 13 days**

- **Backward Pass**

- **T3:** $LF = 13 \rightarrow LS = 9$
- **T2:** $LF = LS(T3) = 9 \rightarrow LS = 3$
- **T1:** $LF = LS(T2) = 3 \rightarrow LS = 0$
- **Slack**
- All tasks have **0 slack** \rightarrow they are all critical.
- **Critical Path = T1 \rightarrow T2 \rightarrow T3 (Total = 13 days)**

PERT Calculation (Handling Uncertainty)

Expected Duration Formula:

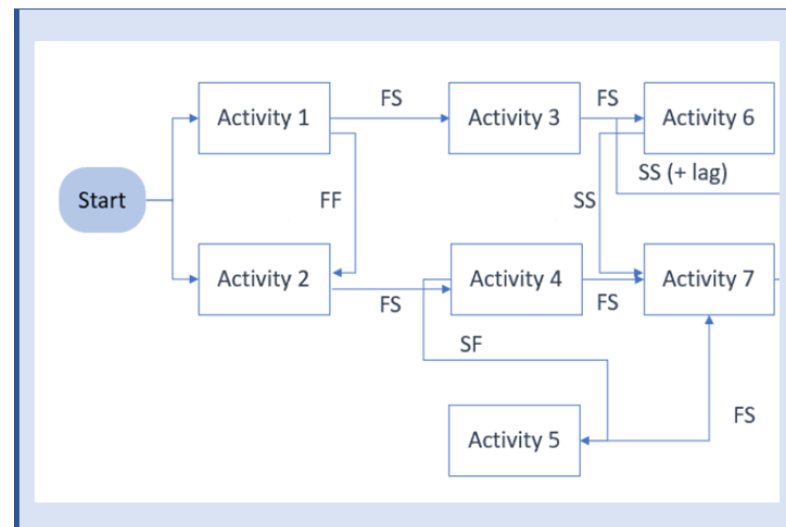
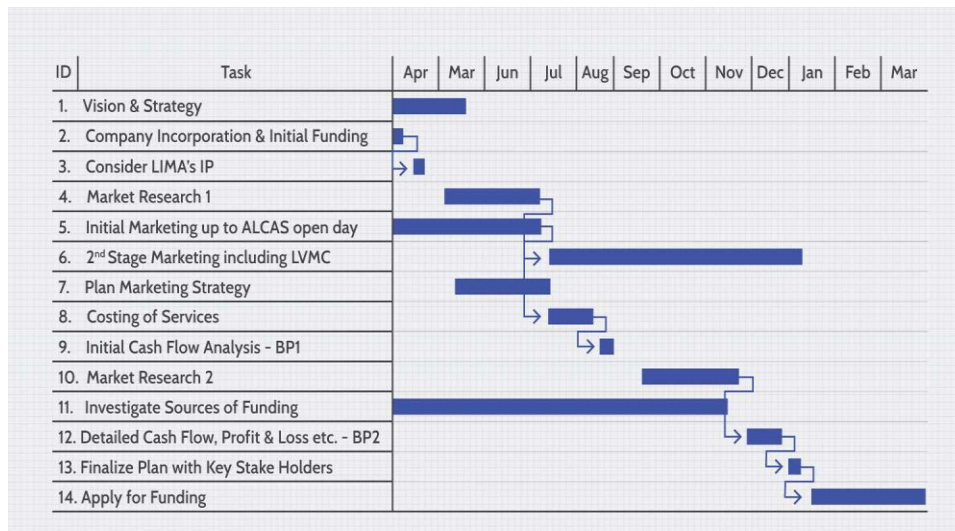
$$TE = \frac{O + 4M + P}{6}$$

Task	Optimistic (O)	Most Likely (M)	Pessimistic (P)
T1	2	3	5
T2	4	6	10
T3	3	4	7

- **Calculate TE for Tasks:**
 - **T1:** $(2 + 4 \times 3 + 5) / 6 = 3.2$ days
 - **T2:** $(4 + 4 \times 6 + 10) / 6 = 6.3$ days
 - **T3:** $(3 + 4 \times 4 + 7) / 6 = 4.0$ days
- **Recalculate CPM using PERT durations**
 - T1 EF = 3.2
 - T2 EF = 3.2 + 6.3 = 9.5
 - T3 EF = 9.5 + 4.0 = 13.5
- **PERT-based project duration ≈ 13.5 days**

Project Management

Gantt Chart and Network Diagram



Source: <https://www.investopedia.com/terms/g/gantt-chart.asp>

Source: project-management.info/project-schedule-network-diagram/

Example (a bit complex)

Task	Duration	Depends On
T1	4	—
T2	6	T1
T3	5	T1
T4	7	T2
T5	3	T2
T6	8	T3
T7	4	T4, T5
T8	6	T6, T7

- ES = Earliest Start
- EF = Earliest Finish
- LS = Latest Start
- LF = Latest Finish

Forward Pass (Compute ES and EF)

- **T1**

- $ES = 0$

- $EF = 0 + 4 = 4$

- **T2 and T3**

- $T2\ ES = EF(T1) = 4$

- $T3\ ES = EF(T1) = 4$

$$\rightarrow EF = 4 + 6 = \mathbf{10}$$

$$\rightarrow EF = 4 + 5 = \mathbf{9}$$

Task	Duration	Depends On
T1	4	—
T2	6	T1
T3	5	T1
T4	7	T2
T5	3	T2
T6	8	T3
T7	4	T4, T5
T8	6	T6, T7

Forward Pass (Compute ES and EF)

Task	Duration	Depends On
T1	4	—
T2	6	T1
T3	5	T1
T4	7	T2
T5	3	T2
T6	8	T3
T7	4	T4, T5
T8	6	T6, T7

- **T4, T5, T6**

- $T4 \text{ ES} = \text{EF}(T2) = 10 \rightarrow \text{EF} = 10 + 7 = \mathbf{17}$
- $T5 \text{ ES} = \text{EF}(T2) = 10 \rightarrow \text{EF} = 10 + 3 = \mathbf{13}$
- $T6 \text{ ES} = \text{EF}(T3) = 9 \rightarrow \text{EF} = 9 + 8 = \mathbf{17}$

- **T7**

- Depends on T4 (17) and T5 (13)
- $\text{ES} = \max(17, 13) = \mathbf{17}$
- $\text{EF} = 17 + 4 = \mathbf{21}$

Forward Pass (Compute ES and EF)

Task	Duration	Depends On
T1	4	—
T2	6	T1
T3	5	T1
T4	7	T2
T5	3	T2
T6	8	T3
T7	4	T4, T5
T8	6	T6, T7

- **T8**
 - Depends on T6 (17) and T7 (21)
 - $ES = \max(17, 21) = \mathbf{21}$
 - $EF = 21 + 6 = \mathbf{27}$
- T1(4), T2(10), T3(9), T4 (17),
T5 (13), T6 (17), T7 (21), T8(27)
- **Total Project Duration = 27 days**

Backward Pass (Compute LS and LF)

- **T8**
 - $LF = 27$
 - $LS = 27 - 6 = \mathbf{21}$
- **T7**
 - $LF = LS(T8) = 21$
 - $LS = 21 - 4 = \mathbf{17}$
- **T6**
 - T6 \rightarrow successor T8 with $LS = 21$
 - $LS = 21 - 8 = \mathbf{13}$, $LF = 21$

Task	Duration	Depends On
T1	4	—
T2	6	T1
T3	5	T1
T4	7	T2
T5	3	T2
T6	8	T3
T7	4	T4, T5
T8	6	T6, T7

Backward Pass (Compute LS and LF)

- **T4, T5**
 - T4 → successor T7 with LS = 17
 - $LS = 17 - 7 = \mathbf{10}$, LF = 17
 - T5 → successor T7
 - $LS = 17 - 3 = \mathbf{14}$, LF = 17
- **T3**
 - T3 successor: T6(LS=13)
 - LF = 13
 - $LS = 13 - 5 = \mathbf{8}$

Task	Duration	Depends On
T1	4	—
T2	6	T1
T3	5	T1
T4	7	T2
T5	3	T2
T6	8	T3
T7	4	T4, T5
T8	6	T6, T7

Backward Pass (Compute LS and LF)

Task	Duration	Depends On
T1	4	—
T2	6	T1
T3	5	T1
T4	7	T2
T5	3	T2
T6	8	T3
T7	4	T4, T5
T8	6	T6, T7

- **T2**
 - T2 successors:
 - T4(LS=10), T5(LS=14) → choose the MIN
 - $LF = 10$
 - $LS = 10 - 6 = 4$
- **T1**
 - Successors: T2(LS=4) and T3(LS=8) → $MIN = 4$
 - $LF = 4$
 - $LS = 4 - 4 = \mathbf{0}$

Slack Calculation ($\text{Slack} = \text{LS} - \text{ES}$)

- ES = Earliest Start
- EF = Earliest Finish
- LS = Latest Start
- LF = Latest Finish
- If $\text{Slack} = 0$,
 - the task is on the critical path; any delay in this task will delay the whole project.
- If $\text{Slack} > 0$,
 - the task has flexibility, and it can be delayed by that many days without affecting project completion.

Slack Calculation ($\text{Slack} = \text{LS} - \text{ES}$)

Task	ES	EF	LS	LF	Slack
T1	0	4	0	4	0 → critical
T2	4	10	4	10	0 → critical
T3	4	9	8	13	4 → can be delayed 4 days
T4	10	17	10	17	0 → critical
T5	10	13	14	17	4 → can be delayed 4 days
T6	9	17	13	21	4 → can be delayed 4 days
T7	17	21	17	21	0 → critical
T8	21	27	21	27	0 → critical

Interpretation:

T1, T2, T4, T7, T8 are critical ($\text{Slack} = 0$) → must finish on time.

T3, T5, T6 have 4 days slack → you could delay each of them up to 4 days without delaying project completion.

Automated / Heuristic Scheduling Approaches

- Advanced search-based techniques in Software Engineering (SBSE) use metaheuristics like
 - Genetic algorithms
 - Particle swarm optimization
 - Ant colony optimization

Factors Affecting Scheduling

- Team skills & availability
- Task complexity
- Requirements clarity
- Technology maturity
- Inter-task dependencies
- External constraints (vendors, clients)

Challenges in Time Scheduling

- Projects face unstable parameters, environmental changes, and competing goals, making manual scheduling inefficient.
- Limited research addresses dynamic models or large-scale uncertainties, with gaps in interactive and adaptive approaches.
- Techniques like GA may not guarantee global optima, and computational complexity can hinder application.

Modern Tools Supporting Scheduling

- Jira
- Microsoft Project
- Trello
- Asana
- OpenProject
- GanttProject

End of Criteria (Time) for Project Evaluation