

SWE 4803

Lec - 5

Quiz



A project aims to develop a cloud-based financial management application for small businesses. The project charter, created by the project manager on behalf of the project sponsor, has been approved. Which of the following processes should be performed next?

- Develop Project Management Plan
- Identify Risks
- Identify Stakeholders
- Develop Project Charter

PLANNING PROCESS GROUP





PLANNING PROCESS GROUP

- 1. Develop Project Management Plan ⇒ Project Integration Management
- 2. Plan Scope Management \Rightarrow Project Scope Management
- 3. Collect Requirements ⇒ Project Scope Management
- 4. Define Scope ⇒ Project Scope Management
- 5. Create WBS ⇒ Project Scope Management
- 6. Plan Schedule Management ⇒ Project Schedule Management
- 7. Define Activities ⇒ Project Schedule Management
- 8. Sequence Activities ⇒ Project Schedule Management
- 9. Estimate Activity Durations ⇒ Project Schedule Management
- 10. Develop Schedule ⇒ Project Schedule Management
- 11. Plan Cost Management ⇒ Project Cost Management
- 12. Estimate Costs ⇒ Project Cost Management

PLANNING PROCESS GROUP



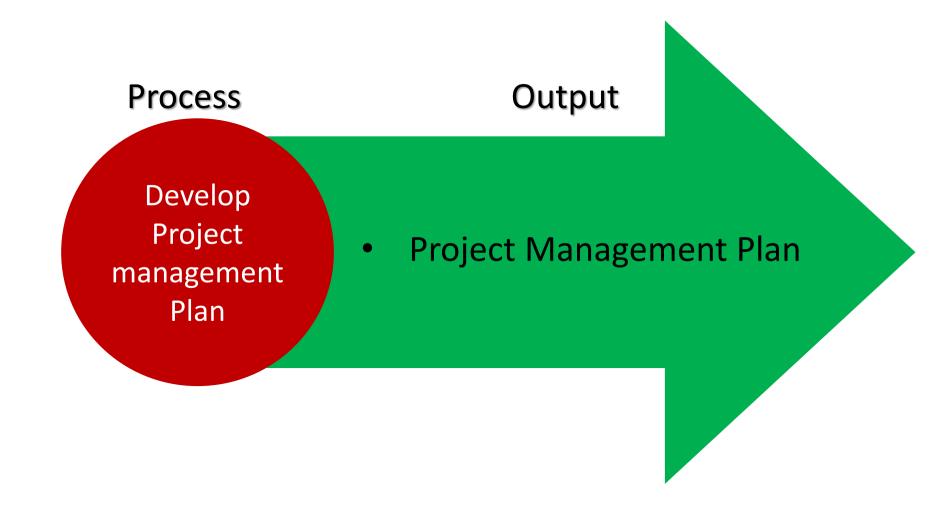


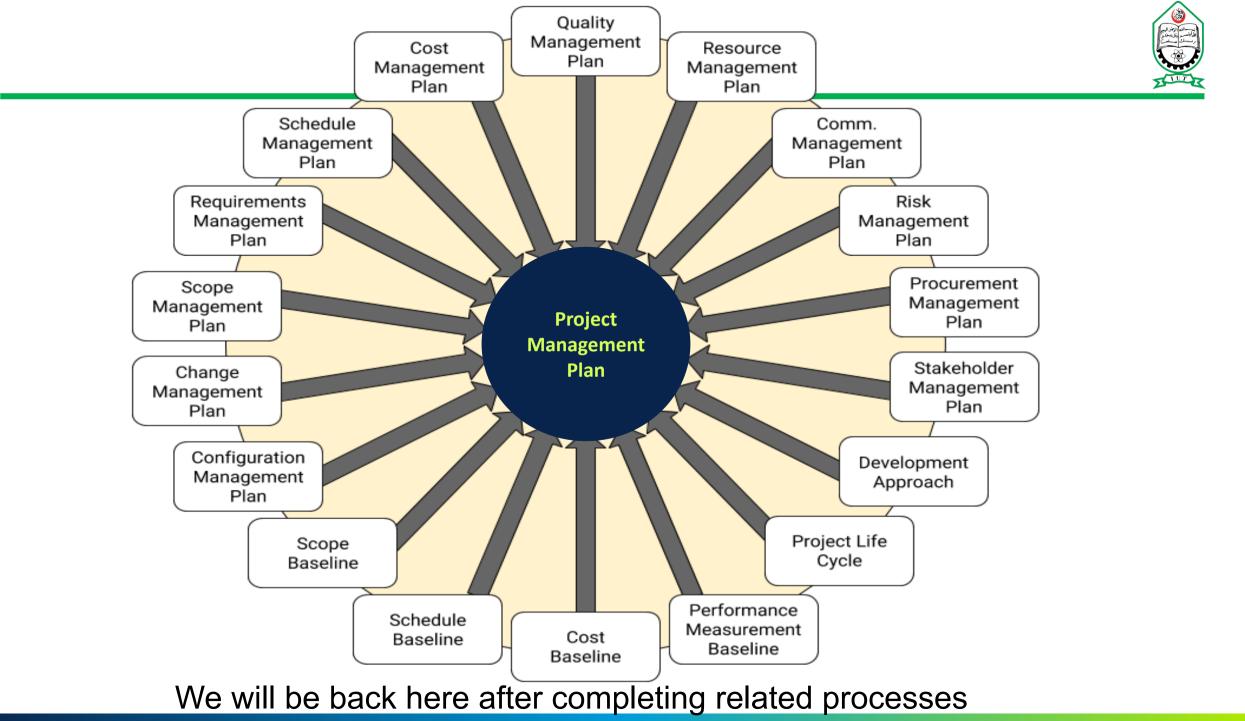
PLANNING PROCESS GROUP

- 13. Determine Budget ⇒ Project Cost Management
- 14. Plan Quality Management ⇒ Project Quality Management
- 15. Plan Resource Management ⇒ Project Resource Management
- 16. Estimate Activity Resources⇒ Project Resource Management
- 17. Plan Communications Management ⇒ Project Communication Management
- 18. Plan Risk Management ⇒ Project Risk Management
- 19. Identify Risks ⇒ Project Risk Management
- 20. Perform Qualitative Risk Analysis ⇒ Project Risk Management
- 21. Perform Quantitative Risk Analysis ⇒ Project Risk Management
- 22. Plan Risk Responses ⇒ Project Risk Management
- 23. Plan Procurement Management ⇒ Project Procurement Management
- 24. Plan Stakeholder Engagement ⇒ Project Stakeholder Management

1. DEVELOP PROJECT MANAGEMENT PLAN

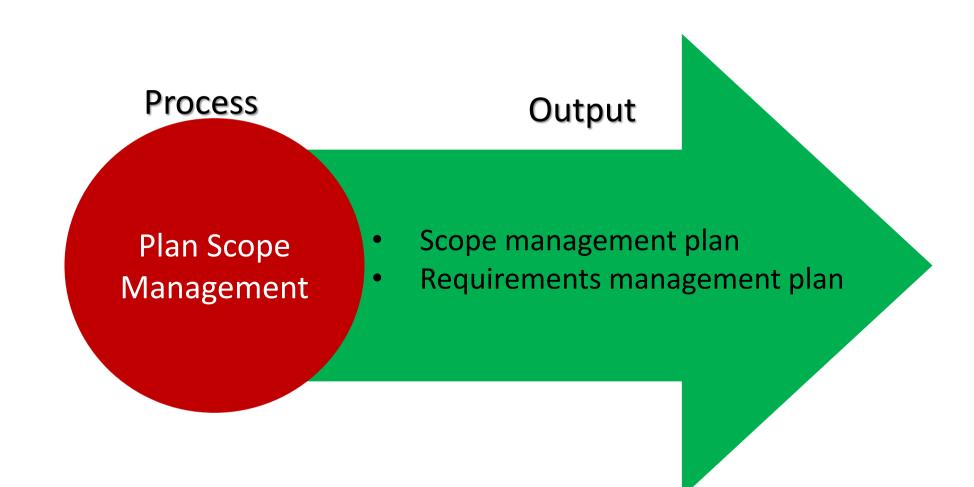








Plan Scope Management is the process of creating a scope management plan that documents how the project and product scope will be **defined**, **validated**, **and controlled**..





KEY CONCEPTS FOR PROJECT SCOPE MANAGEMENT

In the project context, the term "scope" can refer to:

Product scope - The features and functions that characterize a product, service, or result.

Project scope - The work performed to deliver a product, service, or result with the specified features and functions. The term "project scope" is sometimes viewed as including product scope.



Plan Scope Management

Inputs

- .1 Project charter
- .2 Project management plan
 - · Quality management plan
 - Project life cycle description
 - Development approach
- .3 Enterprise environmental factors
- .4 Organizational process assets

Tools & Techniques

- .1 Expert judgment
- .2 Data analysis
 - Alternatives analysis
- .3 Meetings

Outputs

- .1 Scope management plan
- .2 Requirements management plan



Input:

Project charter

• From project charter, we get project purpose, high-level (birds eye view) project <u>description</u>, <u>assumptions</u>, <u>constraints</u>, and high-level <u>requirements</u> that the project is intended to satisfy.

Project Mgt plan

Quality management plan, Project life cycle description, Development approach

EEF & OPA

- EEF Organizations culture, infrastructure, personnel administration and marketplace condition
- OPA Policy, procedure, historical information, lesson learned



Tools & Techniques:

Expert judgment

• From knowledgeable and experienced parties

Data Analysis

• Various ways of collecting requirements, elaborating the project and product scope, creating the product, validating the scope, and controlling the scope are evaluated.

Meetings

 Attendees at these meeting my include project manager, project sponsor, selected team members etc.



Output:

Scope management plan

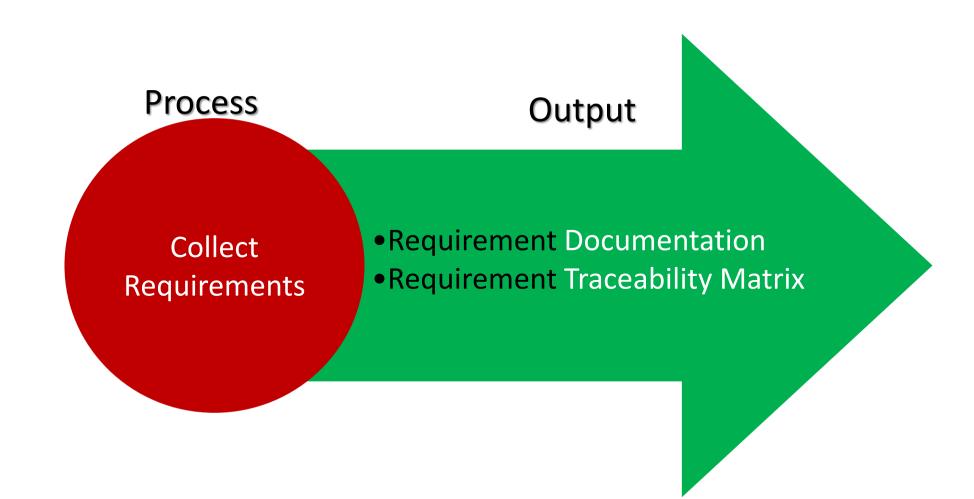
- is a component of the project management plan that describes how the scope will be defined, developed, monitored, controlled, and validated. It includes: (ddmcv)
 - Process for preparing a project scope statement;
 - Process that enables the creation of the WBS from the detailed project scope statement;
 - Process that establishes how the scope baseline will be approved and maintained; and
 - Process that specifies how formal acceptance of the completed project deliverables will be obtained.

Requirement management plan

- How requirements activities will be planned, tracked, and reported; (ptr)
- Configuration management activities such as: how changes will be initiated; how impacts will be analyzed; how they will be traced, tracked, and reported; as well as the authorization levels required to approve these changes;
- Requirements prioritization process;
- Metrics that will be used and the rationale for using them; and
- Traceability structure that reflects the requirement attributes captured on the traceability matrix.



Collect Requirements is the process of <u>determining</u>, <u>documenting</u>, and <u>managing</u> (ddm) stakeholder needs and requirements to meet objectives.





Collect Requirements

Inputs

- .1 Project charter
- .2 Project management plan
 - Scope management plan
 - Requirements management plan
 - Stakeholder engagement plan
- .3 Project documents
 - Assumption log
 - · Lessons learned register
 - Stakeholder register
- .4 Business documents
 - Business case
- .5 Agreements
- .6 Enterprise environmental factors
- .7 Organizational process assets

Tools & Techniques

- .1 Expert judgment
- .2 Data gathering
 - Brainstorming
 - Interviews
 - Focus groups
 - Questionnaires and surveys
 - Benchmarking
- .3 Data analysis
 - Document analysis
- .4 Decision making
 - Voting
 - Multicriteria decision analysis
- .5 Data representation
 - Affinity diagrams
 - Mind mapping
- .6 Interpersonal and team skills
 - Nominal group technique
 - Observation/conversation
 - Facilitation
- .7 Context diagram
- .8 Prototypes

Outputs

- .1 Requirements documentation
- .2 Requirements traceability matrix



Input:

Project charter

• The project charter documents the high-level project description and high-level requirements that will be used to develop detailed requirements.

Project Mgt plan

- Scope management plan.
- Requirement management plan
- Stakeholder management plan

Project Documents

- Assumption log
- Lesson learned register
- Stakeholder register



Input:

Business Documents

• A business document that can influence the Collect Requirements process is the business case, which can describe required, desired, and optional criteria for meeting the business needs.

Agreements

• Agreements can contain project and product requirements.

EEF & OPA

- EEF Organization's culture, Infrastructure, Personnel administration, and Marketplace conditions.
- OPA Policies and procedures, and Historical information and lessons learned repository with information from previous projects.



Tools & Techniques: Expert judgment

Expertise should be considered from individuals or groups with specialized knowledge or training in the following topics:

- Business analysis
- Requirements elicitation
- Requirements analysis
- Requirements documentation
- Project requirements in previous similar projects
- Diagramming techniques
- Facilitation, and
- Conflict management.



Tools & Techniques: Data Gathering

includes:

- Brainstorming
- Interviews
- Questionnaires
- Surveys
- Benchmarking
- Focus groups



Tools & Techniques: Data Analysis

There are wide range of documents that may be analyzed to help elicit relevant requirements.

- 1. Agreements;
- 2. Business plans;
- 3. Business process or interface documentation;
- 4. Business rules repositories;
- 5. Current process flows;
- 6. Marketing literature;
- 7. Problem/issue logs;
- 8. Policies and procedures;
- 9. Regulatory documentation such as laws, codes, or ordinances, etc.;
- 10. Requests for proposal; and
- 11.Use cases.



Tools & Techniques: Decision-making

- ✓ Voting
 - Unanimity: 100% agreement
 - Majority: 50% +
 - Plurality: largest block
- ✓ Autocratic Decision Making
- ✓ Multicriteria decision analysis



Tools & Techniques: Decision-making

Multicriteria decision analysis

- 1. Define your objective
- 2. Define your <u>criteria</u> (non-beneficial & beneficial)
- 3. Determine the weight of each criterion
- 4. List your choices
- 5. Determine your performance values
- 6. Rate your choices
- 7. Normalize the performance values (non-beneficial & beneficial) for non-beneficial criteria, you should take the lowest value of these criteria and divide it by each performance value. For beneficial criteria, you divide each by the maximum value.
- 8. Multiply values by weight
- 9. Calculate the performance scores



Tools & Techniques: Decision-making

Multicriteria decision analysis

To finish the 360-degree image viewer development for your ongoing software project, you need to incorporate an image stitching library. Due to your company's lack of expertise in image stitching, you have decided to outsource the image stitching library and have gathered data on four different image stitching libraries from four different companies. Your job is to assess and select the most suitable image stitching library from four available options with proper justification by considering the criteria provided in Table below. To decide based on the given criteria, you need to consider that the *Resolution Limit* holds the highest weightage of 40%, while all other criteria are considered equally important. Furthermore, price is the only **non-beneficial** factor.

	Criteria				
	Price	Library Size	Resolution Limit	Performance	
Image Stitching Lib 1	\$2500	16 MB	12 MP	Excellent (5)	
Image Stitching Lib 2	\$2000	16 MB	8 MP	Average (3)	
Image Stitching Lib 3	\$3000	32 MB	16 MP	Good (4)	
Image Stitching Lib 4	\$2750	32 MB	8 MP	Good (4)	



Tools & Techniques: Decision-making: Multicriteria decision analysis

		Criteria				
	Price	Library Size	Resolution Limit	Performance		
Image Stitching Lib 1	\$2500	16 MB	12 MP	Excellent (5)		
Image Stitching Lib 2	\$2000	16 MB	8 MP	Average (3)		
Image Stitching Lib 3	\$3000	32 MB	16 MP	Good (4)		
Image Stitching Lib 4	\$2750	32 MB	8 MP	Good (4)		

Normalizing

	Criteria				
	Price	Library Size	Resolution Limit	Performance	
Image Stitching Lib 1	(2000/2500)	16/32	12 /16	5/5	
Image Stitching Lib 2	(2000/2000)	16 /32	8 /16	3/5	
Image Stitching Lib 3	(2000/3000)	32 /32	16 /16	4/5	
Image Stitching Lib 4	(2000/2750)	32 /32	8/16	4/5	

0.76 0.62 0.8934 0.7054



Tools & Techniques: Decision-making: Multicriteria decision analysis

	Criteria				
	Price	Library Size	Resolution Limit	Performance	
Image Stitching Lib 1	\$2500	16 MB	12 MP	Excellent (5)	
Image Stitching Lib 2	\$2000	16 MB	8 MP	Average (3)	
Image Stitching Lib 3	\$3000	32 MB	16 MP	Good (4)	
Image Stitching Lib 4	\$2750	32 MB	8 MP	Good (4)	

Multiply by weights

		Criteria			
	Price	Library Size	Resolution Limit	Performance	
Image Stitching Lib 1	.8*.2	.5*.2	.75*.4	1*.2	
Image Stitching Lib 2	1*.2	.5*.2	.5*.4	.6*.2	
Image Stitching Lib 3	.667*.2	1*.2	1*.4	.8*.2	
Image Stitching Lib 4	.727*.2	1*.2	.5*.4	.8*.2	

0.76

0.62

0.8934

0.7054



Tools & Techniques: Decision-making: Multicriteria decision analysis

	Criteria				
	Price	Library Size	Resolution Limit	Performance	
Image Stitching Lib 1	\$2500	16 MB	12 MP	Excellent (5)	
Image Stitching Lib 2	\$2000	16 MB	8 MP	Average (3)	
Image Stitching Lib 3	\$3000	32 MB	16 MP	Good (4)	
Image Stitching Lib 4	\$2750	32 MB	8 MP	Good (4)	

Multiply by weights

	Criteria				
	Price	Library Size	Resolution Limit	Performance	
Image Stitching Lib 1	.16	.1	.3	.2	
Image Stitching Lib 2	.2	.1	.2	.12	
Image Stitching Lib 3	.1334	.2	.4	.16	
Image Stitching Lib 4	.1454	.2	.2	.16	

0.76 0.62 0.8934 0.7054

SOLVE NOW



You are leading a project to implement a **real-time sentiment analysis module** for a customer service platform. To speed up development and ensure reliability, your team has decided to integrate an **external Natural Language Processing (NLP) API** instead of building the sentiment engine from scratch.

You have shortlisted **four different NLP APIs** from different vendors. Your task is to evaluate and select the most suitable NLP API using the criteria listed in the table below. The **Accuracy Rate** is considered the most important factor, carrying a **weight of 40%**, while the other criteria (Latency, API Flexibility, Documentation Quality, and Cost) are equally weighted. **Cost** is the only **non-beneficial** criterion whereas all others are beneficial.

API Option	Accuracy Rate (%)	Latency (ms)	API Flexibility (1–10)	Documentation Quality (1–10)	Cost (\$/month)
API A	91	280	8	7	450
API B	87	200	6	9	400
API C	93	320	7	8	550
API D	89	250	9	6	380



Tools & Techniques: Data representation

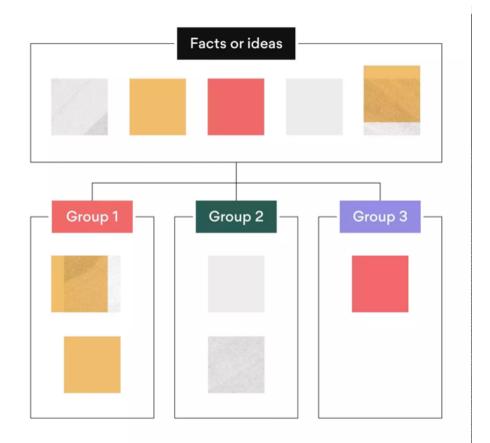
✓ **Affinity diagrams.** Affinity diagrams allow large numbers of ideas to be classified into groups for <u>review</u> and <u>analysis</u>.

✓ **Mind mapping.** Mind mapping consolidates ideas created through individual brainstorming sessions into a single map to reflect commonality and differences in understanding and to generate new ideas.



Tools & Techniques: Data representation

✓ Affinity diagrams.



Challenges in Production

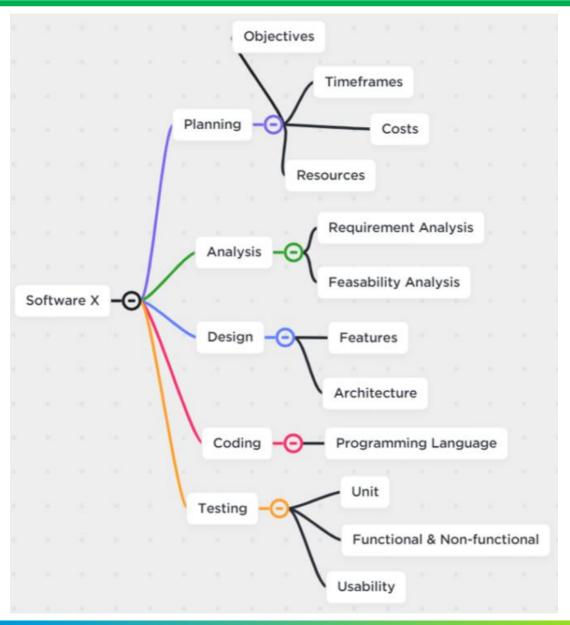
Staff	Distribution	Quality	Capacity
Lack of Staff training	Not enugh trucks.	Varialble ingredients quality	Insufficient ovens
Difficulties recruiting	Cooling systems in trucks unreliable.	Packaging not strong Enough	Limited storage space
High Overtime	Project damaged in transit		Seasonal demand



Tools & Techniques: Data representation

✓ Mind mapping.

To build a mind map diagram, you first choose a **central idea** or main topic. The main topic will have branches that connect it to other linked **subtopics**, and you repeat the process until you have a literal map of your mind!





Tools & Techniques: Data representation

✓ Mind mapping.

Prepare a mind map for **Arranging a Meeting**



Tools & Techniques: Interpersonal and Team skills

- ✓ Nominal group technique
- ✓ Observation and conversation
- √ facilitation



Tools & Techniques: Context diagram

- √ Example of a scope model
- ✓ Visually depict the product scope by showing business system and how people and other system interact with it.

Tools & Techniques: Prototype

✓ Prototyping is a method of obtaining early feedback on requirements by providing a model of the expected product before actually building it.



Output:

Requirement Documentation

- Business requirement
- Stakeholders' requirements
- Solution requirements
- Functional & non-functional requirements
- Transition and readiness requirements
- Project requirement
- Quality requirement

Requirement traceability matrix

- business need, opportunity, goal
- Project objectives
- WBS deliverables
- Product design and development
- Test scenario
- High level to detail level

REQUIREMENTS TRACEABILITY MATRIX

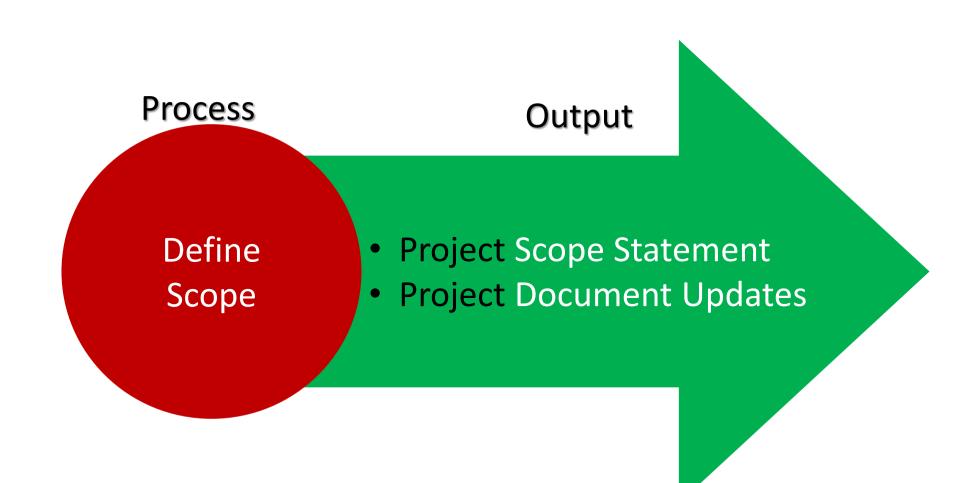
Project Name: Online Flight Booking Application

Business Requirements Document BRD		Functional Requirements Document FSD			Test Case Document
Business Requirement ID#	Business Requirement / Business Use case	Functional Requirement ID#	Functional Requirement / Use Case	Priority	Test Case ID#
BR_1	Reservation Module	FR_1	One Way Ticket booking	High	TC#001 TC#002
		FR_2	Round Way Ticket		TC#003 TC#004
		FR_3	Multicity Ticket booking	High	TC#005 TC#006
BR_2	Payment Module	FR_4	By Credit Card	High	TC#007 TC#008
		FR_5	By Debit Card	High	TC#009
		FR_6	By Reward Points	Medium	TC#010 TC#011

DEFINE SCOPE



The process of developing a detailed description of the project and product



DEFINE SCOPE



Define Scope

Inputs

- .1 Project charter
- .2 Project management plan
 - · Scope management plan
- .3 Project documents
 - Assumption log
 - Requirements documentation
 - Risk register
- .4 Enterprise environmental factors
- .5 Organizational process assets

Tools & Techniques

- .1 Expert judgment
- .2 Data analysis
 - Alternatives analysis
- .3 Decision making
 - Multicriteria decision analysis
- .4 Interpersonal and team skills
 - Facilitation
- .5 Product analysis

Outputs

- .1 Project scope statement
- .2 Project documents updates
 - Assumption log
 - Requirements documentation
 - Requirements traceability matrix
 - Stakeholder register



Tools & Techniques: Product Analysis

- ✓ product analysis techniques include but are not limited to:
 - Product <u>Features</u> and <u>Functionality</u>,
 - Requirements analysis,
 - Systems analysis,
 - Lifecycle Analysis,
 - Value analysis, and
 - Value engineering.



Output:

Project Scope Statement

- Product scope description.
- Deliverables
- Acceptance criteria
- Project exclusion

Project
Document
Updates

- Assumption log
- Requirement documentation
- Requirement tractability matrix
- Stakeholder register

The project charter contains high-level information, while the project scope statement contains a detailed description of the scope components.



Acceptance criteria:

- ✓ **Acceptance criteria** are a set of **conditions or requirements** that a product, feature, or user story must meet to be accepted by the user, customer, or product owner.
- ✓ They answer the question: "What does "done" mean for this work?"

Purpose of Acceptance Criteria

- Define **clear expectations** for a feature or task
- Help developers and testers understand what to build and test
- Avoid misunderstandings
- Guide user story validation

In Agile (like Scrum), acceptance criteria are often written **before development starts** as part of the **Definition of Done**.

Example (To-Do List App)

User Story:

As a user, I want to mark a task as complete, so that I can track what I've finished.

Acceptance Criteria:

- User can click a checkbox to mark the task complete.
- Completed tasks appear with a strikethrough.
- Completed tasks are saved even after refreshing the page.
- A completed task can be marked as "incomplete" again.



Assumption log:

✓ **Assumption log** is a document or list that records all the **assumptions** made during a project. These are things the team believes to be true **without full proof or certainty** at the time of planning or execution.

What Is an Assumption?

An **assumption** is something you accept as true **for now**, to plan or make progress, even though it might change later.

Example:

"We assume the third-party API will be available 24/7."

"We assume the client will provide all required data by next week."

Purpose of the Assumption Log

- Track all assumptions in one place
- Allow the team to review, monitor, and update them
- Help identify **risks** if assumptions turn out to be false
- Improve decision-making during uncertainty



Assumption log:

ID	Assumption	Date Logged	Owner	Status
A1	Users will access the system only during business hours	2025-06-17	Project Manager	Valid
A2	Client will approve the design within 5 working days	2025-06-17	Product Owner	Pending
A3	No major changes in regulatory requirements	2025-06-17	Compliance Officer	To be reviewed

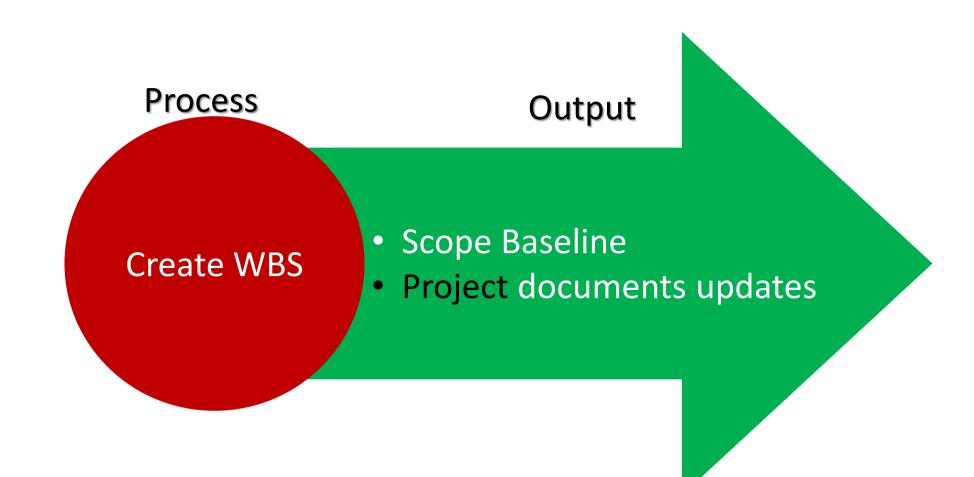


Assumption log:

ID	Assumption	Status	Meaning
A1	API will be stable during development	Valid	Already tested; it's working fine.
A2	Client will provide data by Friday	Pending	Client has not confirmed yet.
A3	Only desktop users will access the platform	At Risk	Mobile usage stats are increasing - assumption may be invalid soon.
A4	Government regulations won't change this year	To Be Reviewed	May need to check recent updates in policy.
A5	Project team will remain the same until delivery	Closed	Project delivered, assumption no longer relevant.



- ✓ Work Breakdown Structure (WBS)
- ✓ The process of developing a detailed description of the project and product





The process of subdividing project deliverables and project work into smaller, more manageable components.

Create WBS

Inputs

- .1 Project management plan
 - Scope management plan
- .2 Project documents
 - Project scope statement
 - Requirements documentation
- .3 Enterprise environmental factors
- .4 Organizational process assets

Tools & Techniques

- .1 Expert judgment
- .2 Decomposition

Outputs

- .1 Scope baseline
- .2 Project documents updates
 - · Assumption log
 - Requirements documentation

WORK BREAKDOWN STRUCTURE (WBS)



✓ The WBS represents a logical decomposition of the work to be performed and focuses on how the product, service, or result is naturally subdivided. It is an outline of what work is to be performed.

LEVEL	ELEMENT DESCRIPTION	
1	Project	
2	Category	
3	Subcategory	
4	Sub-Subcategory	
5	Work Package	



Input:

PROJECT MANAGEMENT PLAN

• A project management plan component includes but is not limited to the scope management plan.

PROJECT DOCUMENTS

- Scope management plan.
- Requirement documentation
- Stakeholder management plan

EEF & OPA

- EEF industry-specific WBS standards
- OPA Policies, procedures, and templates for the WBS, Project files from previous projects, Lessons learned from previous projects.



Tools & Techniques:

- 1. Expert Judgement
- 2. Decomposition

Decomposition activities:

- Identifying and analyzing the deliverables and related work
- Structuring and <u>organizing</u> the WBS
- Decomposing the upper WBS levels into lower-level detailed components
- Developing and assigning identification codes to the WBS components
- Verifying that the degree of decomposition of the deliverables is appropriate.

Rolling wave planning and 100% rule applies in case of decomposition



Output:

Project Scope Baseline

- The **scope baseline** is the approved version of a scope statement, WBS, and its associated WBS dictionary, which can be changed only through **formal change control procedures** and is used as a basis for comparison
- scope Statement
- WBS
- Work Package
- Planning Package
- WBS Dictionary
 - √ Code of account identifier
 - ✓ Description of work
 - ✓ Responsible organization
 - ✓ Schedule milestones, associated activities
 - ✓ Resource required, cost estimation,
 - ✓ acceptance criteria

Project Document Updates

- Assumption log
- Requirement documentation



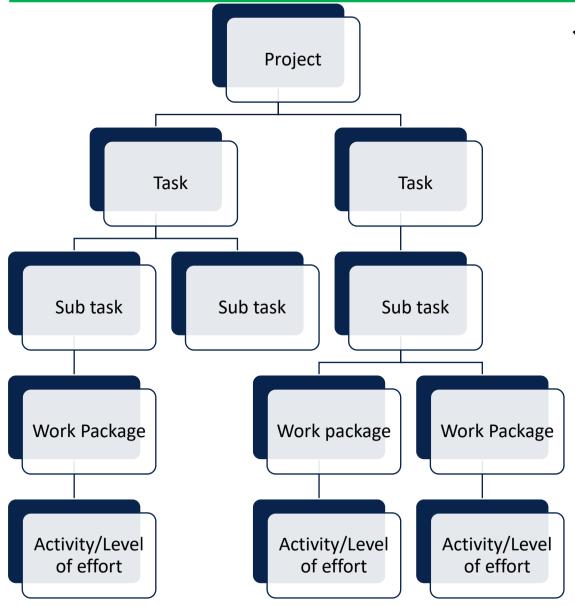
Output:

WBS Dictionary Element	Description		
Code of Account Identifier	Unique alphanumeric code assigned to the work package for tracking and referencing.		
Description of Work	Detailed explanation of the work scope, objectives, and deliverables for the WBS element.		
Responsible Organization	Team, department, or individual accountable for performing the work.		
Schedule Milestones & Activities	Key dates and checkpoints, including start/end dates and associated activities needed.		
Resources Required & Cost Estimation	Types and quantities of resources (labor, materials, equipment) and estimated costs.		
Acceptance Criteria	Standards or conditions that define when the work is considered complete and acceptable.		



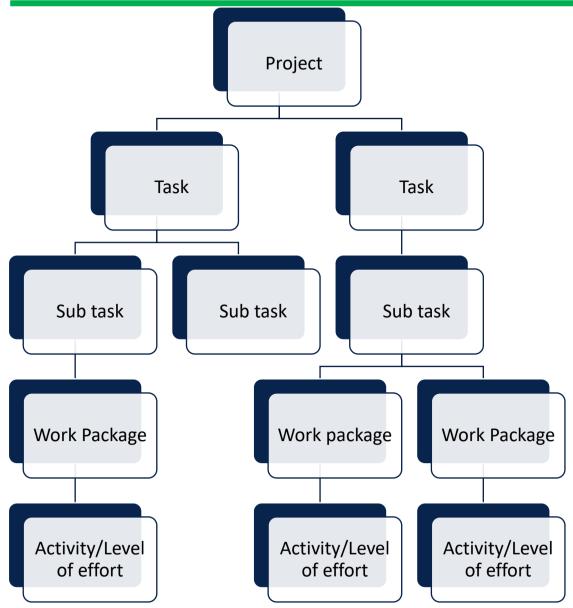
- ✓ WBS is used for every project
- ✓ WBS does not show the sequence of the execution. That will be determined when a schedule is developed.
- ✓ The concept behind the WBS is very simple.
- ✓ It requires you to break down complex tasks into smaller tasks until you reach a point where you can no longer subdivide the tasks.





- ✓ Tasks should be broken down until you are able to:
 - ✓ Manage the task
 - ✓ Estimate the time required to complete the task
 - ✓ Estimate the cost involved with the task
 - ✓ Rule 1 "80-hour rule" which means that no single activity or group of activities at the lowest level of detail of the WBS to produce a single deliverable should be more than 80 hours of effort.





- ✓ Tasks should be broken down until you are able to:
 - ✓ Manage the task
 - ✓ Estimate the time required to complete the task
 - ✓ Estimate the cost involved with the task
 - ✓ Rule 2 The second rule of thumb is that no activity or group of activities at the lowest level of detail of the WBS should be longer than a single reporting period. Thus, if the project team is reporting progress monthly, then no single activity or series of activities should be longer than one month long.



Example: Developing a Login Feature for a Mobile App

Step 1: Define the Deliverable

Deliverable: Login Feature Module

Backend Authentication (90 hours, 8 days) is too big according to Rule 1 and Rule 2.

Break it into smaller activities:

WBS Activity	Estimated Effort (Hours)	Duration (Days)	Notes
1. Design Login UI	40	3	Under 80 hours & within 1 reporting period (weekly report assumed)
2. Develop Frontend Logic	70	5	Under 80 hours & within 1 reporting period
3. Develop Backend Authentication	90	8	Break down required (exceeds 80 hours and 1 reporting period)
4. Integration Testing	35	3	Under 80 hours and 1 reporting period
5. Bug Fixing	30	2	Under 80 hours and 1 reporting period



Example: Developing a Login Feature for a Mobile App

Step 1: Define the Deliverable

Deliverable: Login Feature Module

Backend Authentication (90 hours, 8 days) is too big according to Rule 1 and Rule 2.

Break it into smaller activities:

Activity	Estimated Effort (Hours)	Duration (Days)
1. Design Login UI	40	3
2. Develop Frontend Logic	70	5
3.1 Setup Database Authentication	40	3
3.2 Implement API Endpoints	30	2
3.3 Security Testing & Validation	20	2
4. Integration Testing	35	3
5. Bug Fixing	30	2



Example: Developing a Login Feature for a Mobile App

Step 1: Define the Deliverable

Deliverable: Login Feature Module

Backend Authentication (90 hours, 8 days) is too big according to Rule 1 and Rule 2.

Break it into smaller activities:

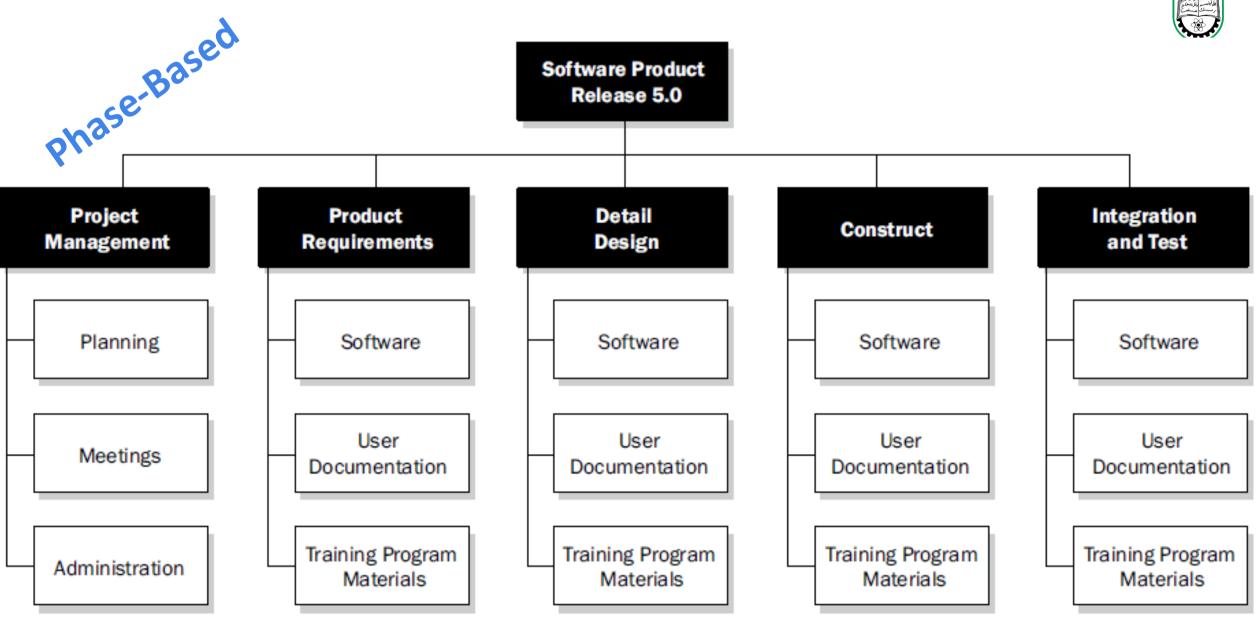
Sub-activity	Estimated Effort (Hours)	Duration (Days)	Notes
3.1 Setup Database Authentication	40	3	Meets both rules
3.2 Implement API Endpoints	30	2	Meets both rules
3.3 Security Testing & Validation	20	2	Meets both rules

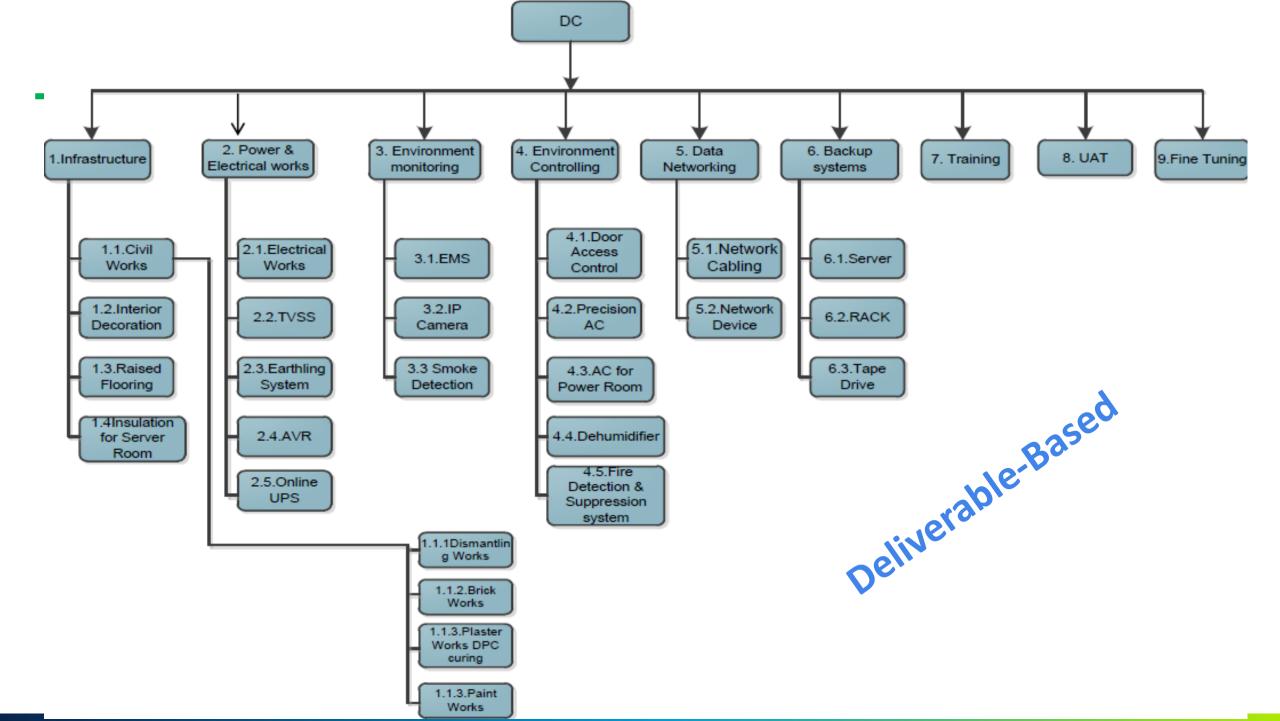
MORE ON CREATE WBS



- ✓ WBS Development Guidelines:
 - ✓ Stop breaking down work when you reach a low enough level to do an estimate of the desired accuracy
 - ✓ Involve those people who must do the work in the planning process
 - ✓ WBS should be completed before the schedule
 - ✓ WBS does not have to be symmetrical
- ✓ There are two types of **WBS**:
 - Deliverable-Based
 - Phase-Based.

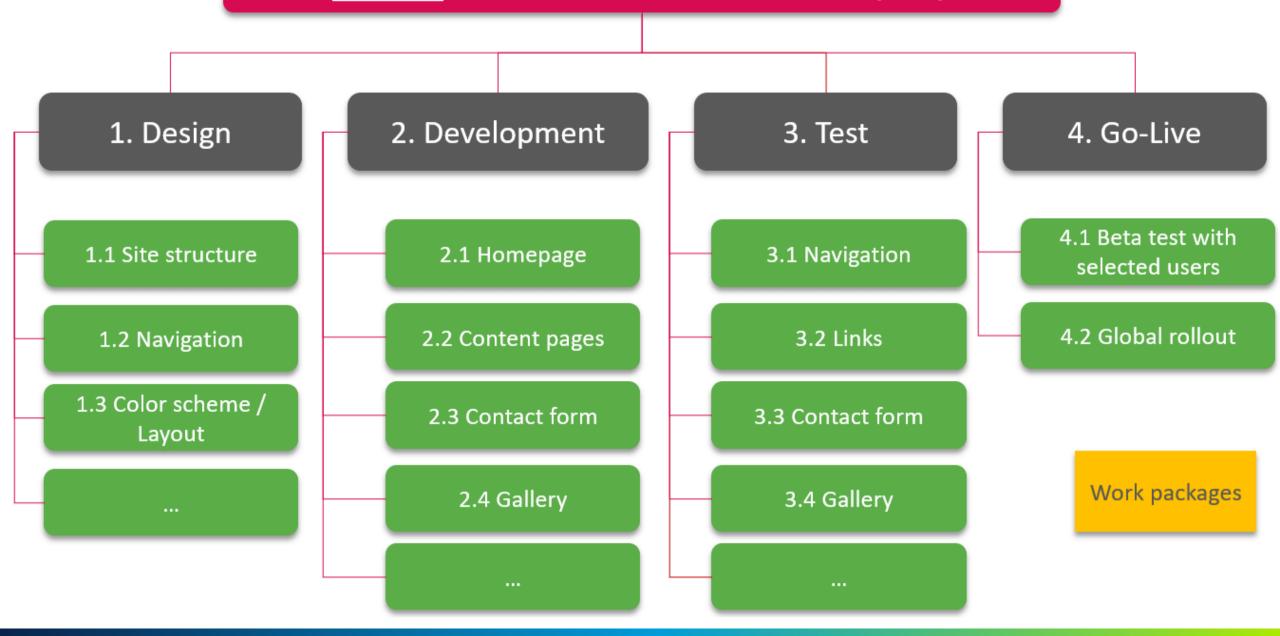








Project: New Website for Our Company

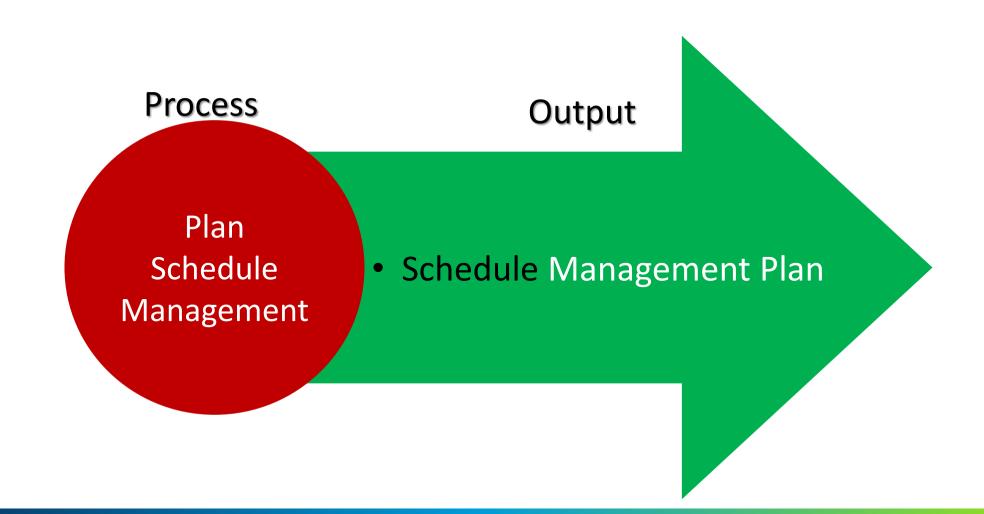


Project: New Website for Our Company

3. Interactive elements 1. Homepage 2. Subpages 1.1 Photo slider 2.1 Text pages German 3.1 Contact page 1.2 Introductory texts, 2.2 Text pages English 3.2 Photo gallery pages company presentation 1.3 Pictograms and 2.3 Profile pages for 3.3 Blog further links the team 1.4 News area 3.4 Chat 3.5 Social media functions



The process of establishing the <u>policies</u>, <u>procedures</u>, and <u>documentation</u> for planning, developing, managing, executing, and controlling the project schedule.





It provides guidance and direction on how the project schedule will be managed throughout the project

Plan Schedule Management

Inputs

- .1 Project charter
- .2 Project management plan
 - Scope management plan
 - · Development approach
- .3 Enterprise environmental factors
- .4 Organizational process assets

Tools & Techniques

- .1 Expert judgment
- .2 Data analysis
- .3 Meetings

Outputs

.1 Schedule management plan



Input:

PROJECT CHARTER

• defines the summary milestone schedule that will influence the management of the project schedule.

PROJECT MANAGEMENT PLAN

- Scope management plan how the scope will be defined and developed
- Development approach help to define the scheduling approach, estimating techniques, scheduling tools, and techniques for controlling the schedule.

EEF & OPA

- EEF Organizational culture and structure, physical resource availability,
 Scheduling software, Commercial databases, such as standardized estimating data
- OPA Historical information and lessons learned repositories, Existing formal and informal schedule development, management- and control-related policies, procedures, and guidelines, Templates and forms, Monitoring and reporting tools.



Tools & Techniques:

Expert judgment

- Expertise should be considered from individuals or groups with specialized knowledge or training in previous, similar projects:
 - ✓ Schedule development, management, and control;
 - ✓ Scheduling methodologies (e.g., predictive or adaptive life cycle);
 - ✓ Scheduling software; and
 - ✓ The specific industry for which the project is developed.

Data Analysis

• Alternatives analysis can include determining which schedule methodology to use, or how to combine various methods on the project.

Meetings

may hold planning meetings to develop the schedule management plan.

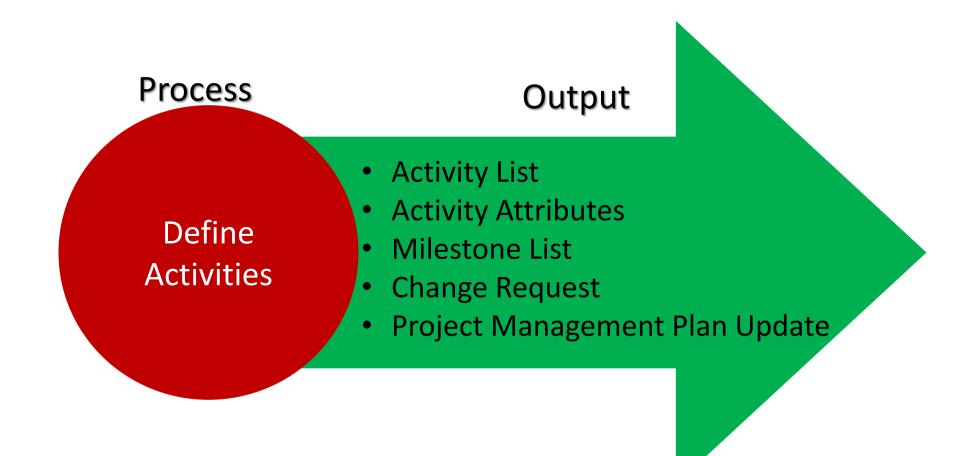


Output: Schedule management plan

- 1. Schedule Model: Scheduling methodologies and scheduling tools
- 2. Release and iteration length. When using an adaptive life cycle, the time-boxed periods for releases, waves, and iterations are specified.
- 3. Prescribed level of details necessary to manage the work
- 4. Level of accuracy acceptable range used to determine activity duration
- **5. Unit of measure**: staff hour, meter, liters, tons, kilometers etc.
- 6. Control threshold: variance threshold
- 7. Rule of performance measurement: Earned value measurement technique (baseline, fixed formula) schedule performance measurement
- 8. Reporting formats: format and frequency
- 9. Process descriptions: schedule management processes are documented.



The process of identifying and documenting the specific actions to be performed to produce the project deliverables.





it decomposes work packages into schedule activities that provide a basis for <u>estimating</u>, <u>scheduling</u>, <u>executing</u>, <u>monitoring</u>, and <u>controlling</u> the project <u>work</u>.

Define Activities

Inputs

- .1 Project management plan
 - Schedule management plan
 - Scope baseline
- .2 Enterprise environmental factors
- .3 Organizational process assets

Tools & Techniques

- .1 Expert judgment
- .2 Decomposition
- .3 Rolling wave planning
- .4 Meetings

Outputs

- .1 Activity list
- .2 Activity attributes
- .3 Milestone list
- .4 Change requests
- .5 Project management plan updates
 - Schedule baseline
 - Cost baseline



Input:

PROJECT MANAGEMENT PLAN

- Scope management plan defines the schedule methodology, the duration of waves for rolling wave planning, and the level of detail necessary to manage the work
- Scope baseline- The project WBS, deliverables, constraints, and assumptions documented in the scope baseline are considered explicitly while defining activities.

EEF & OPA

- EEF Organizational culture and structure, Project management information system (PMIS), Commercial databases
- OPA Historical information and lessons learned repositories, Standardized processes, Templates that contain a standard activity list, Existing formal and informal activity planning-related policies, procedures, and guidelines,



Tools & Techniques:

- ✓ Expert Judgement
- ✓ Decomposition
- ✓ Rolling Wave planning
- ✓ Meeting



Output:

Activity List

- Comprehensive list including all schedule activities
- Included activity identifies and scope of work description

Activity Attributes • the activity identifier, activity codes, activity description, predecessor activities, successor activities, logical relationships, leads and lags, resource requirements, imposed dates, constraints and assumptions

Milestone List

• Identifies all the milestones and indicates whether the milestone is mandatory (required by the contract) or optional (based on project requirements)



Output:

Change Request

• Once the project has been baselined, the progressive elaboration of deliverables into activities may reveal work that was not initially part of the project baselines.

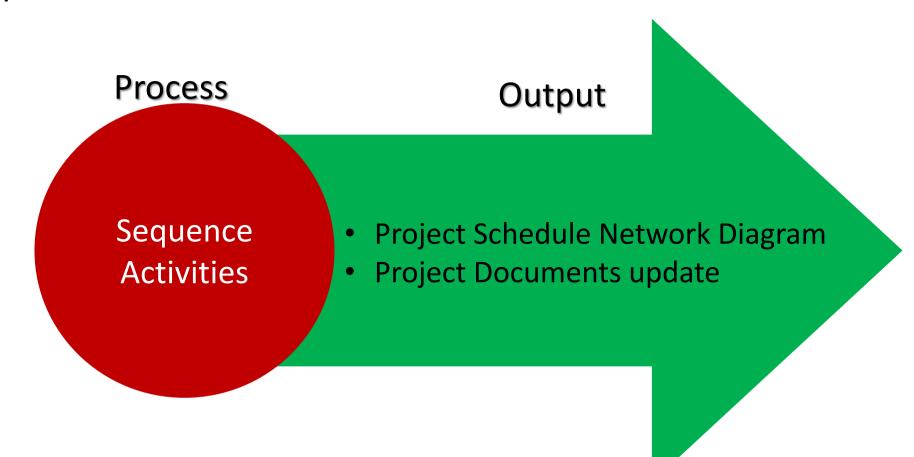
Project Management Plan Update

- Schedule baseline
- Cost baseline

SEQUENCE ACTIVITIES



- ✓ process of identifying and documenting relationships among the project activities.
- ✓ it defines the logical sequence of work to obtain the greatest efficiency given all project constraints.





The Sequence Activities process concentrates on converting the project activities from a list to a diagram to act as a first step to publish the schedule baseline.

Sequence Activities

Inputs

- .1 Project management plan
 - Schedule management plan
 - Scope baseline
- .2 Project documents
 - Activity attributes
 - Activity list
 - Assumption log
 - Milestone list
- .3 Enterprise environmental factors
- .4 Organizational process assets

Tools & Techniques

- .1 Precedence diagramming method
- .2 Dependency determination and integration
- .3 Leads and lags
- .4 Project management information system

Outputs

- .1 Project schedule network diagrams
- .2 Project documents updates
 - Activity attributes
 - Activity list
 - Assumption log
 - Milestone list



Input:

PROJECT MANAGEMEN T PLAN

- Schedule Management Plan
- Scope Baseline

PROJECT DOCUMENTS

- Activity List, Activity Attribute, Milestone list
- Assumption log

EEF & OPA

- EEF Government or industry standards, Project management information system (PMIS), Scheduling tools, and Organization work authorization systems.
- OPA Standard, PMIS, tools, work authorization system, Scheduling methodology, LL knowledge base



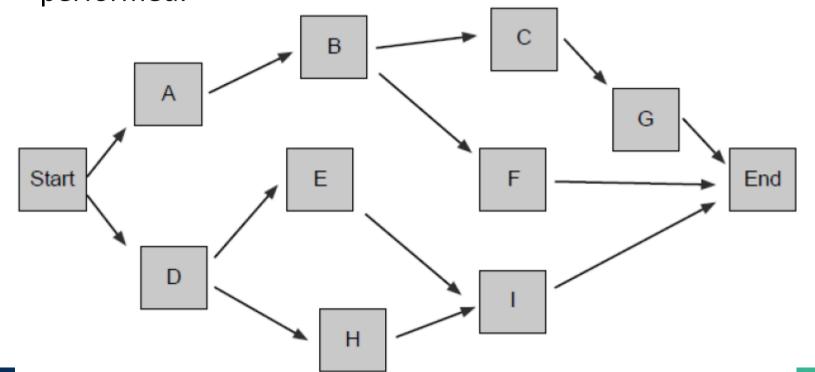
Tools & Techniques:

- ✓ precedence diagramming method (PDM)
- ✓ Dependency Determination
- ✓ Leads and Lags
- ✓ Project management information system (PMIS)



Tools & Techniques: precedence diagramming method (PDM)

✓ is a technique used for constructing a schedule model in which activities are represented by nodes and are graphically linked by one or more logical relationships to show the sequence in which the activities are to be performed.



Name	Predecessor
Start	-
Α	Start
В	А
С	В
D	Start
E	D
F	В
G	С
Н	D
I	E,H
End	F,G,I



Tools & Techniques: precedence diagramming method (PDM)

Finish to Start (F-S)

The initiation of the successor activity depends upon the completion of the predecessor activity

you cannot start testing a method before the test cases are developed

Start-to-start (S-S)

The initiation of the successor activity depends upon the initiation of the predecessor activity

Frontend coding can start only after UI design has started..

Types of Dependency

(based on logical relationship)

Finish-to-finish (F-F)

The completion of the successor activity depends upon the completion of the predecessor activity

The research report is finalized when data analysis is complete.

Start-to-finish (S-F)

The completion of the successor activity depends upon the initiation of the predecessor activity

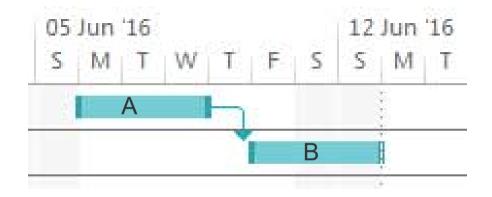
Database Migration- to stop the old system you must start the new system



Tools & Techniques: precedence diagramming method (PDM)



Finish to Start (F-S)



Task B cannot start until task A finishes.

A – Write User Manual

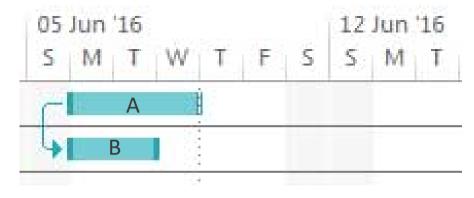
B – Print User Manual



Tools & Techniques: precedence diagramming method (PDM)



Start to Start (S-S)



task B can only start when task A has started

A – Write Code for S/W Module X

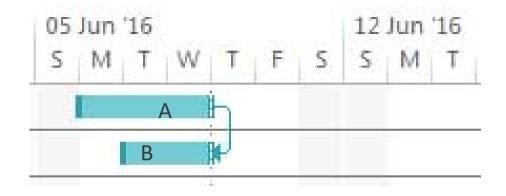
B – Write DB Scripts for S/W Module X



Tools & Techniques: precedence diagramming method (PDM)



Finish to Finish (F-F)



task B cannot finish until task A finishes

A – Write Code for S/W Module X

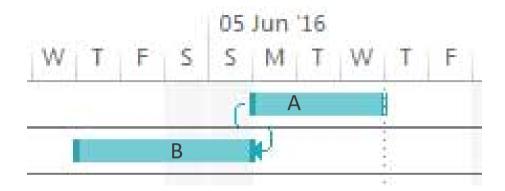
B – Unit Test S/W Module X



Tools & Techniques: precedence diagramming method (PDM)



Start to Finish (S-F)



task B cannot finish until task A starts.

A – Start using New Software System

B – Phase out Old Software System



Tools & Techniques: precedence diagramming method (PDM)

- ✓ FS is the most commonly used type of precedence relationship. The SF relationship is very rarely used
- ✓ Two activities can have two logical relationships at the same time (for example, SS and FF).
- ✓ Multiple relationships between the same activities are not recommended, so a decision has to be made to select the relationship with the highest impact.
- ✓ Closed loops are also not recommended in logical relationships.



Tools & Techniques: precedence diagramming method (PDM)

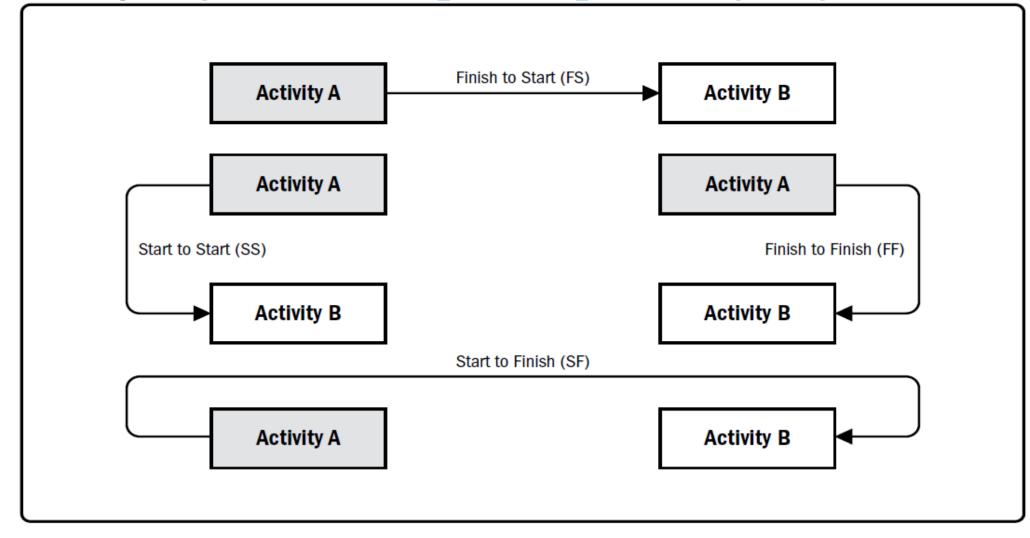


Figure 6-9. Precedence Diagramming Method (PDM) Relationship Types

Example of Precedence Diagramming Method



For one of your project followings are the task list:

- 1. technical design of module A (duration: 10 days),
- 2. technical design of module B (duration: 5 days),
- development of module A (duration: 15 days),
- 4. development of module B (duration: 20 days), and
- 5. development of feature F in module B (duration: 1 day).

The technical dependencies are:

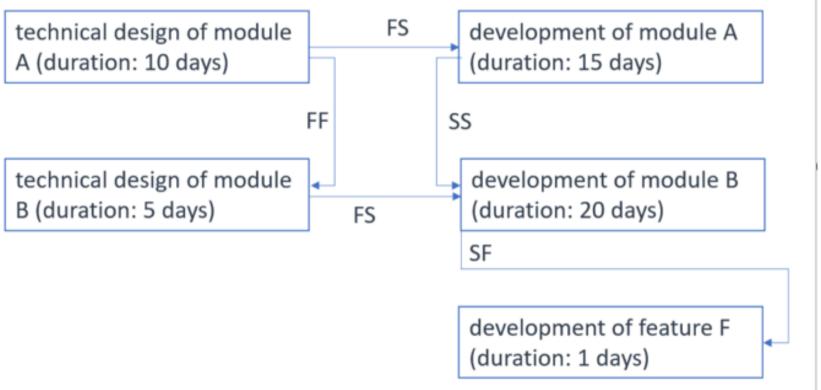
- 1. The technical design of module B cannot finish until the technical design of module A has been completed.
- 2. The technical designs of module A must be completed before the module A development can start.
- 3. The technical designs of module B must be completed before the module B development can start.
- 4. The development of module B can only be started when the development of module A has been started.
- 5. The development of feature F cannot be finished before the development of module B has started (although it can be developed independently, it needs to be integrated into module B)

Logical Relationship:

- 1. Activity (1) is predecessor of activity (2) in a finish-to-finish (FF) dependency (There could also be a start-to-start (SS) dependency between (1) and (2) but the FF relationship is prioritized as its impact is higher)
- 2. Activity (1) is predecessor of activity (3) in a finish-to-start (FS) relationship
- 3. Activity (2) is predecessor of activity (4) in a finish-to-start (FS) relationship
- 4. Activity (3) is predecessor of activity (4) in a start-to-start (SS) relationship
- 5. Activity (5) is a successor of activity (4) and in a start-to-finish (SF) relationship

Example of Precedence Diagramming Method





Logical Relationship:

- 1. Activity (1) is predecessor of activity (2) in a finish-to-finish (FF) dependency (There could also be a start-to-start (SS) dependency between (1) and (2) but the FF relationship is prioritized as its impact is higher)
- 2. Activity (1) is predecessor of activity (3) in a finish-to-start (FS) relationship
- 3. Activity (2) is predecessor of activity (4) in a finish-to-start (FS) relationship
- 4. Activity (3) is predecessor of activity (4) in a start-to-start (SS) relationship
- 5. Activity (5) is a successor of activity (4) and in a start-to-finish (SF) relationship



Tools & Techniques: Dependency Determination

Mandatory dependencies

Contractually required or inherent in the nature of the work (called hard logic)

You must design the database schema before coding queries..

Discretionary dependencies

defined by project management team. Known as "softlogic", "preferential logic" or "preferred logic"

You prefer to finish UI before starting backend, even if both can go in parallel.

Types of dependency

(based on source of dependency)

External dependencies

involve a relationship between project activities and non project activities.

Waiting for regulatory approval before launching a product.

Internal dependencies

Internal dependencies involve a precedence relationship between project activities and are generally inside the project team's control.

The QA team can begin testing only after development is done.



Tools & Techniques: Leads and Lags

- ✓ The amount of time whereby a successor activity can be advanced with respect to a predecessor activity.
- ✓ A Lead provides acceleration to the Successor Activity.

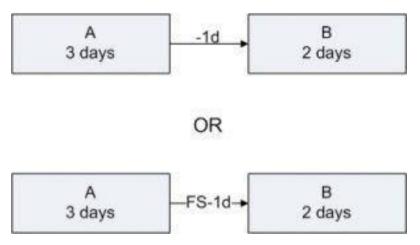
Consider two activities A and B.

Duration of A - 3 days

Duration of B - 2 days

B has a **Finish to Start** Relationship with A with a Lead of 1 day.

The scheduled start of B is 1 day before the scheduled Finish of A.



Project Team would need 4 days to complete these activities. If the said FS relationship was without any Lead, the Project Team would have needed 5 days.



Tools & Techniques: Leads and Lags

- ✓ The amount of time whereby a successor activity is required to be delayed with respect to a predecessor activity.
- ✓ A Lag provides mandatory delay to the Successor Activity.

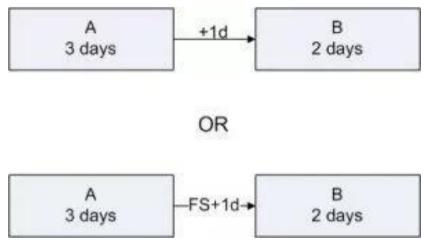
Consider two activities A and B.

Duration of A - 3 days

Duration of B - 2 days

B has a Finish to Start Relationship with A with a Lag of 1 day.

The scheduled Start of B is 1 day after the scheduled Finish of A.



Project Team would need 6 days to complete these activities. If the said FS relationship was without any Lag, the Project Team would have needed 5 days.

EXAMPLE OF LEADS AND LAGS



In a project, the durations are estimated as follows:

- 1. technical design of module A: 10 days
- 2. technical design of module B: 5 days
- 3. development of module A: 15 days
- 4. development of module B: 20 days

The dependencies are:

- 1. the technical designs of module A and B, respectively, must be completed before the respective module can be developed,
- 2. the technical design of module B makes reference to that of module A, thus, the module A design needs to be 50% completed before the module B design can be started, and
- 3. the development of module A depends on module B, which needs to be 25% completed before the team can start the development of module A.

Logical Relationship:

- 1. Activity 1 and activity 2 (SS with lag 5d)
- Activity 1 and activity 3 (FS)
- Activity 2and activity 4 (FS)
- 4. Activity 3 and activity 4 (SS with lag 5d)



1. technical design of module A: 10 days

2. technical design of module B: 5 days

3. development of module A: 15 days

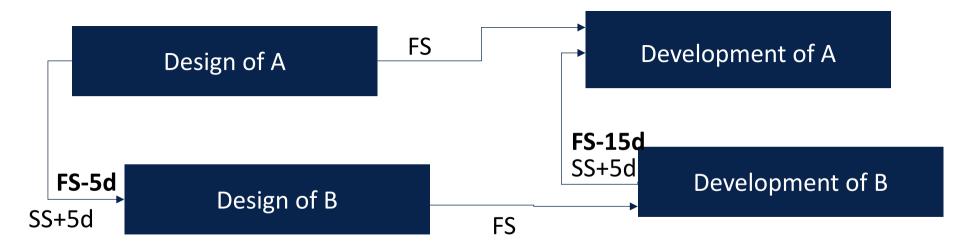
4. development of module B: 20 days

	days									
Activity	5	10	15	20	25	30	35	40	45	50
Design module A										
Design module B										
Develop module B										
Develop module A										

Schedule of the project based on just finish-to-start relationships.



Activity	Dependencies and durations	Lag time	Lead time
Design module A	takes 10 days	first activity	first activity
Design module B	can start 5 days after the design of module A started and takes 5 days	5 days (after start of A)	5 days (before finishing A)
Develop module B	can start when the design of module B is completed and takes 20 days	no lag	no lead
Develop module A	can start when module B is 25% completed and takes 15 days	5 days (after start of B)	15 days (before completion of B)





Activity	Dependencies and durations	Lag time	Lead time
Design module A	takes 10 days	first activity	first activity
Design module B	can start 5 days after the design of module A started and takes 5 days	5 days (after start of A)	5 days (before finishing A)
Develop module B	can start when the design of module B is completed and takes 20 days	no lag	no lead
Develop module A	can start when module B is 25% completed and takes 15 days	5 days (after start of B)	15 days (before completion of B)

	days									
Activity	5	10	15	20	25	30	3 5	40	45	50
Design module A										
Design module B										
Develop module B										
Develop module A										



Tools & Techniques: Project management information systems

✓ is a software system or tool that helps plan, execute, monitor, and control all aspects of a project.

Function	Example
Planning	Create schedules, budgets, task lists
Resource management	Assign people, track availability
Communication	Share updates, send notifications
Documentation	Store project files, reports, meeting notes
Monitoring & control	Track progress, view earned value metrics, manage changes
Reporting	Generate dashboards and performance reports



Output:

Project schedule network diagram

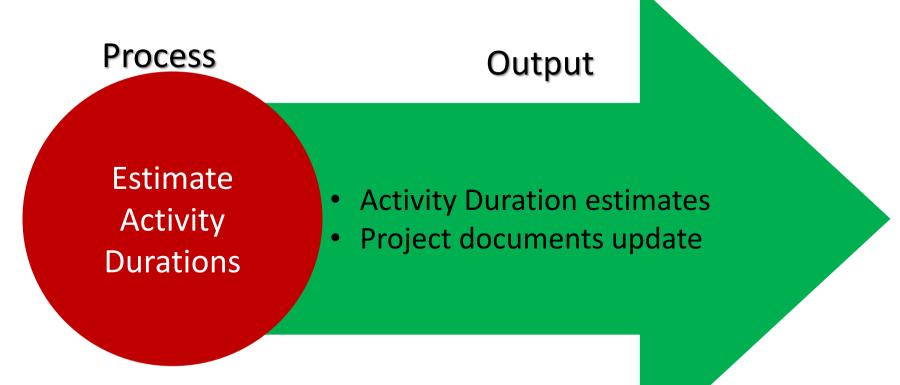
• Schematic displays of the project's activities and their logical relationship

Project Document Update

- Activity attributes
- Activity list.
- Assumption log.
- Milestone list.



- ✓ is the process of estimating the number of work periods needed to complete individual activities with estimated resources.
- ✓ The key benefit of this process is that it provides the amount of time each activity will take to complete.





Estimate Activity Durations

Inputs

- .1 Project management plan
 - Schedule management plan
 - Scope baseline
- .2 Project documents
 - Activity attributes
 - Activity list
 - · Assumption log
 - Lessons learned register
 - Milestone list
 - Project team assignments
 - Resource breakdown structure
 - Resource calendars
 - Resource requirements
 - Risk register
- .3 Enterprise environmental factors
- .4 Organizational process assets

Tools & Techniques

- .1 Expert judgment
- .2 Analogous estimating
- .3 Parametric estimating
- .4 Three-point estimating
- .5 Bottom-up estimating
- .6 Data analysis
 - · Alternatives analysis
 - Reserve analysis
- .7 Decision making
- .8 Meetings

Outputs

- .1 Duration estimates
- .2 Basis of estimates
- .3 Project documents updates
 - Activity attributes
 - Assumption log
 - Lessons learned register



Input:

Project Mgt Plan

- Schedule Management Plan
- Schedule baselines

Project Documents

- Activity attributes
- Activity list, Assumption log, Lessons learned register, Milestone list, Project team assignments, Resource breakdown Structure, Resource calendars, Resource requirements, Risk register

EEF & OPA

- EEF Duration estimating databases and other reference data, Productivity metrics, Published commercial information, and Location of team members.
- OPA Historical duration information, Project calendars, Estimating policies,
 Scheduling methodology, and Lessons learned repository.



Tools & Techniques:

- 1. Expert Judgment
- 2. Analogous Estimating
 - Also called 'Top-Down-Estimating', means using the actual duration of a previous, similar activity as the basis for estimating the duration of a future activity."
 - Frequently used when limited amount of detailed information is available
 - Less costly and time consuming
 - More reliable when previous activities are similar in fact and circumstance and not only in appearance and when the individual preparing the estimate has the experience of similar activities
 - Gross value estimation approach. Adjusted according to the difference
- 3. Parametric estimates
 - Quantity of work multiplied by productivity rate can be used to estimate activity duration



Tools & Techniques:

- 4. Three points estimates
 - To improve the accuracy of estimates
 - ✓ Optimistic (O) Estimate for all favorable conditions with no risks or changes
 - ✓ Pessimistic (P) Estimate for all unfavorable conditions with all negative risks occurring and no mitigation
 - ✓ Most likely (M) Estimate for both favorable and unfavorable conditions, with some risks occurring.

Depending on the assumed distribution of values within the range of the three estimates, the expected duration, tE, can be calculated. One commonly used formula is triangular distribution:

$$E = (O + M + P) / 3.$$

Triangular distribution is used when there is insufficient historical data or when using judgmental data.



Tools & Techniques:

- 5. Bottom-up estimating
 - is a method of estimating project duration or cost by aggregating the estimates of the lower-level components of the WBS.
- 6. Data analysis
- 7. Decision-making
- 8. Meetings



Output:

Duration Estimates

- Duration estimates are quantitative assessments of the likely number of time periods that are required to complete an activity, a phase, or a project.
- A range of 2 weeks ± 2 days, which indicates that the activity will take at least 8 days and not more than 12 (assuming a 5-day work week); or
- A 15% probability of exceeding 3 weeks, which indicates a high probability— 85%—that the activity will take 3 weeks or less.

Basis of estimates

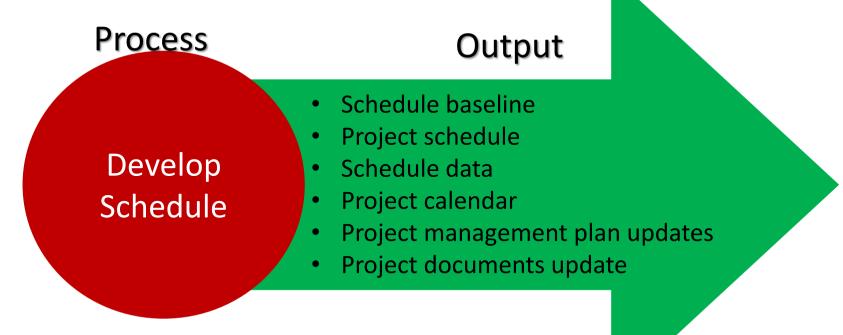
- How it was developed
- Assumption & Constraints
- Confidence level etc.

Project Documents Update

- Activity Attributes
- Assumption log
- Lesson learned register



- ✓ is the process of analyzing activity sequences, durations, resource requirements, and schedule constraints to create a schedule model for project execution and monitoring and controlling.
- ✓ The key benefit of this process is that it generates a schedule model with planned dates for completing project activities..





Develop Schedule

Inputs

- .1 Project management plan
 - Schedule management plan
 - Scope baseline
- .2 Project documents
 - Activity attributes
 - Activity list
 - Assumption log
 - Basis of estimates
 - Duration estimates
 - Lessons learned register
 - Milestone list
 - Project schedule network diagrams
 - Project team assignments
 - Resource calendars
 - Resource requirements
 - Risk register
- .3 Agreements
- .4 Enterprise environmental factors
- .5 Organizational process assets

Tools & Techniques

- .1 Schedule network analysis
- .2 Critical path method
- .3 Resource optimization
- .4 Data analysis
 - What-if scenario analysis
 - Simulation
- .5 Leads and lags
- .6 Schedule compression
- .7 Project management information system
- .8 Agile release planning

Outputs

- .1 Schedule baseline
- .2 Project schedule
- .3 Schedule data
- .4 Project calendars
- .5 Change requests
- .6 Project management plan updates
 - Schedule management plan
 - Cost baseline
- .7 Project documents updates
 - Activity attributes
 - Assumption log
 - Duration estimates
 - Lessons learned register
 - Resource requirements
 - Risk register



Input:

Project Mgt Plan

- Schedule Management Plan
- Scope baselines

Project Documents

 Activity attributes, Activity list, Assumption log, Basis of estimates, Duration estimates, Lessons learned register, Milestone list, Project schedule network diagrams, Project team assignments, Resource calendars, Resource requirements, Risk register

Agreements

• Vendors may have an input to the project schedule as they develop the details of how they will perform the project work to meet contractual commitments.

EEF & OPA

- EEF Government or industry standards, and Communication channels.
- OPA Scheduling methodology containing the policies governing schedule model development and maintenance, and Project calendar(s).



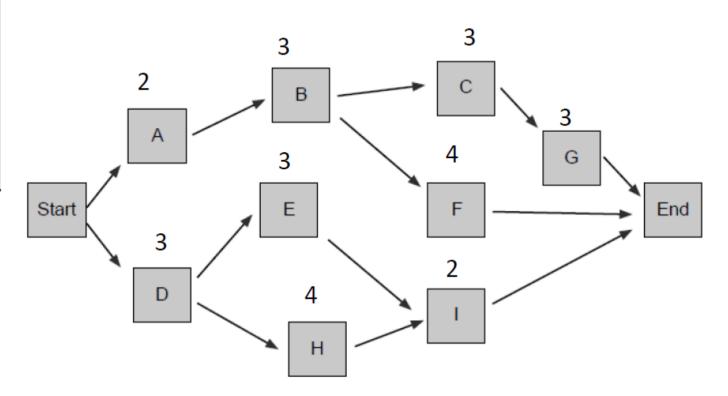
Tools & Techniques:

- 1. Schedule network analysis
 - is the primary technique used to generate the project schedule model. It employs several other techniques such as critical path method, resource optimization techniques, and modeling techniques.
- 2. Critical Path Method
 - A method used to estimate the minimum project duration
 - Determine the schedule flexibility
 - Calculates Early start, early finish, late start and late finish
 - Performs forward and backward pass
 - Without regarded for any resource limitation



Tools & Techniques: Critical Path Method

Name	Predecessor
Start	
Α	Start
В	Α
С	В
D	Start
E	D
F	В
G	С
Н	D
I	E, H
Finish	F, G, I





Tools & Techniques: Critical Path Method

Abbreviations:

ES – Early Start

LS – Late Start

EF – Early Finish

LF – Late Finish

P – Predecessor

S – Successor

CP – Critical Path

D - Duration

Formulas:

0 Method

ESs= max (ESp+ Dp)

 $ES = EF ext{ of } P$

EF = ES + D

LFp = min(LFs - Dp)

LF = LS of S

LS = LF - D

1 Method

ES: EF of Pred+1

EF: (ES + Duration)-1

LF: LS of successor-1

LS: (LF-duration)+1

ES	Dur	EF			
Activity Name					
LS	Float/ Slack	LF			

The critical path is the sequence of activities with the longest duration.

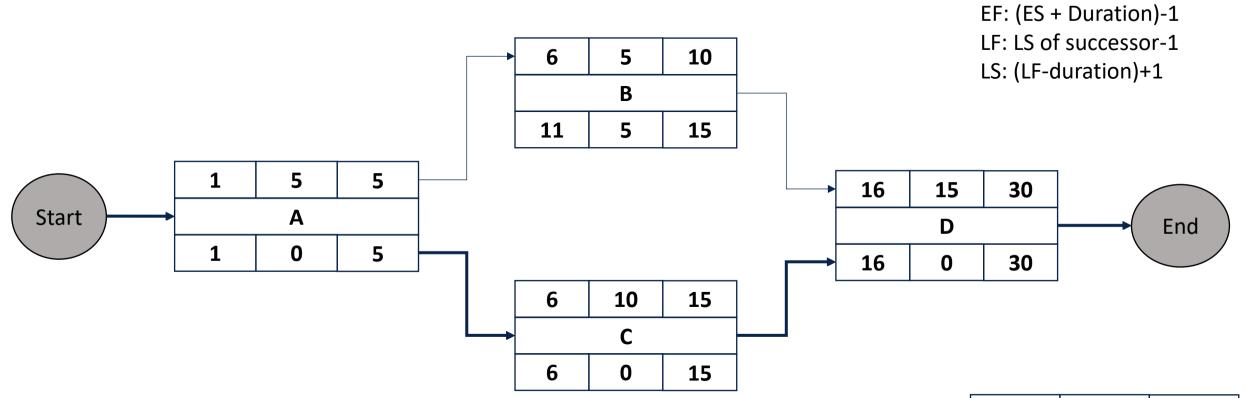


Tools & Techniques: Critical Path Method

Concept	Formula	Interpretation
Free Float Determines how many days you can delay an activity without delaying the ES of the next activity.	Free Float = ES (S) – EF (P)	Number of days this activity can be delayed without delaying the ES of the next activity
Total Float Determines how many days you can delay an activity without delaying the project.	Total Float = LS – ES Total Float = LF – EF	Number of days this activity can be delayed without delaying the project.



Tools & Techniques: Critical Path Method



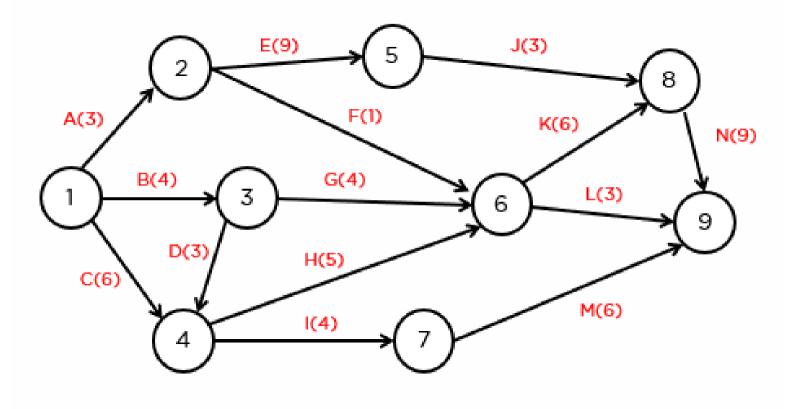
ES	Dur	EF								
Activity Name										
LS	Float/	I F								
LS	Slack	LF								

ES: EF of Pred+1

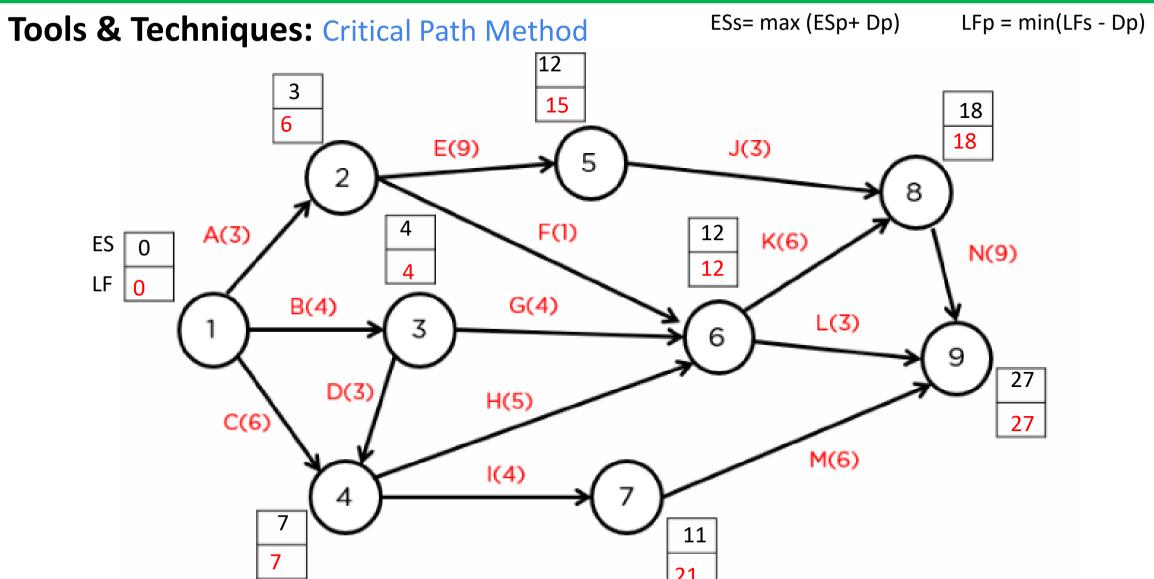


Tools & Techniques: Critical Path Method

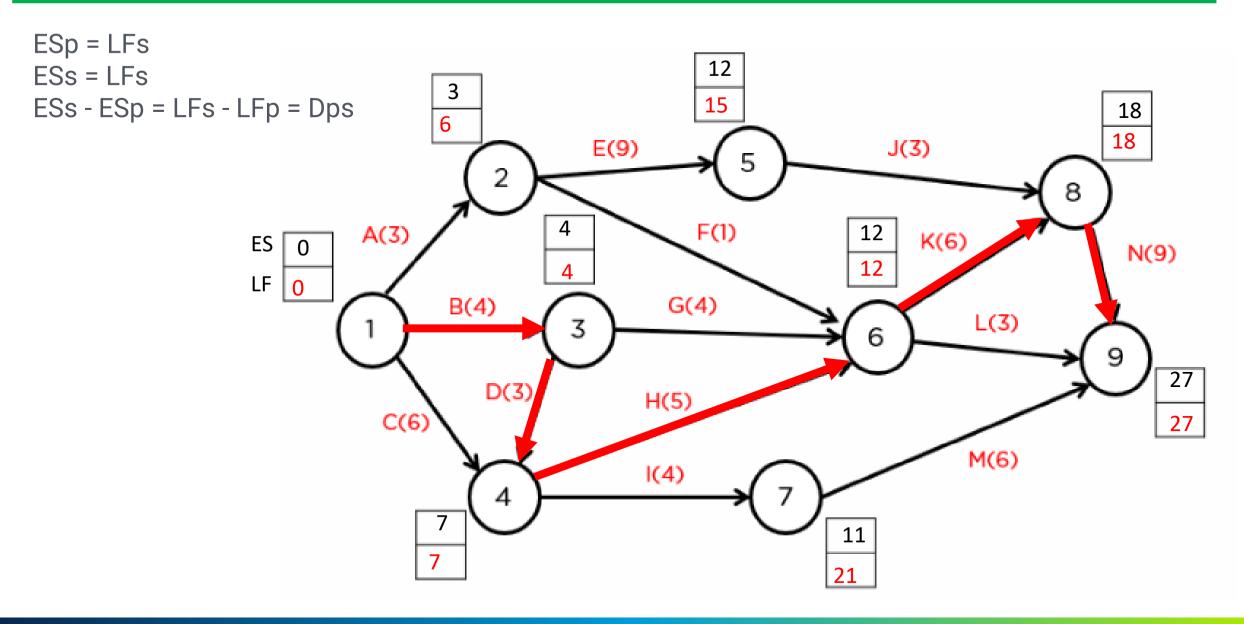
Activity	Immediate Predecessor	Duration (Mon)
А	-	3
В	-	4
С	-	6
D	В	3
Е	Α	9
F	Α	1
G	В	4
Н	C, D	5
I	C, D	4
J	E	3
K	F, G, H	6
L	F, G, H	3
M	1	6
N	J, K	9











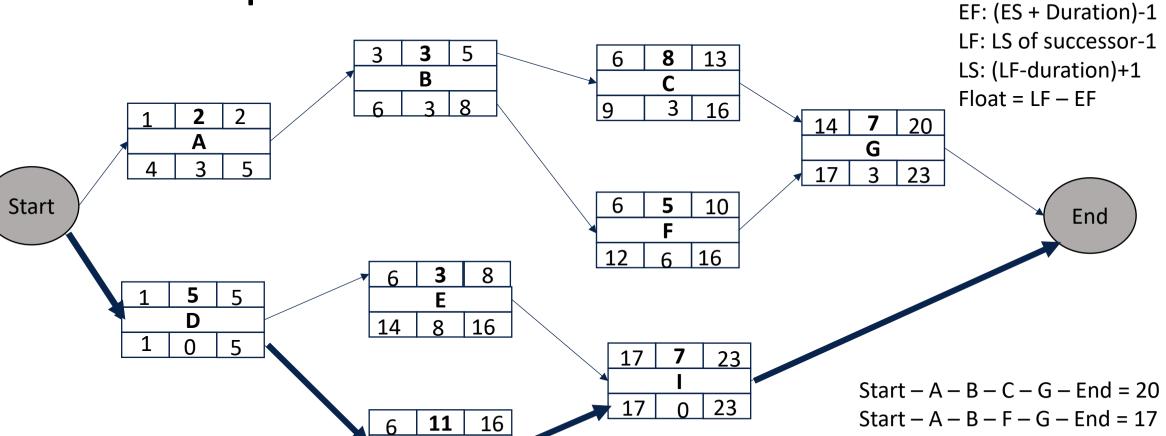


ES: EF of Pred+1

Start - D - E - I - End = 15

Start - D - H - I - End = 23

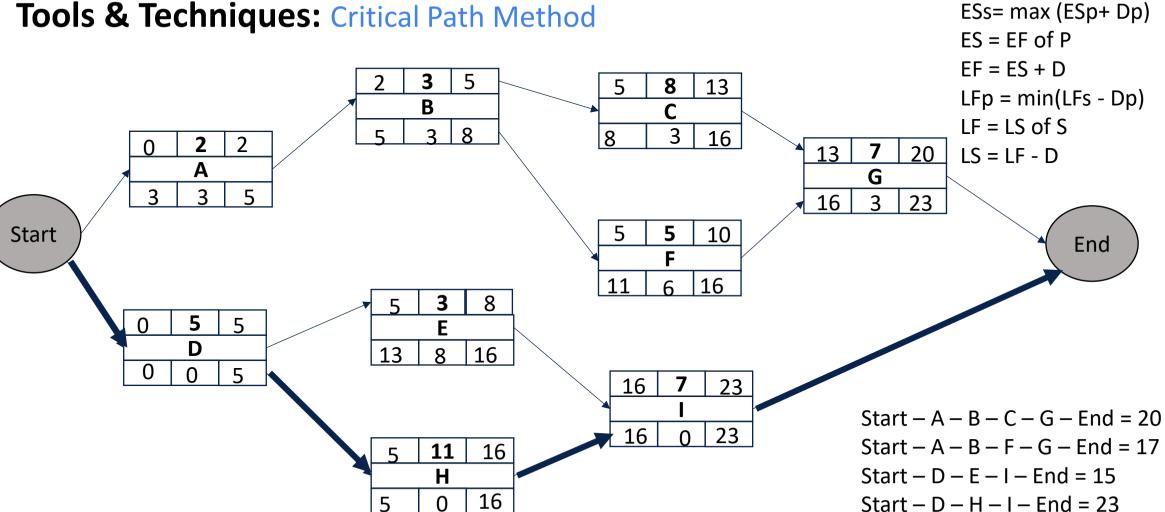




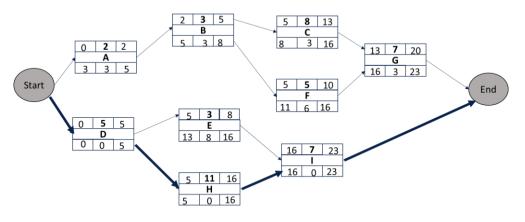
16



Tools & Techniques: Critical Path Method





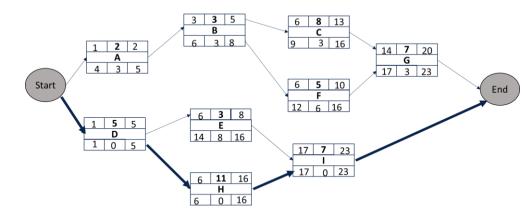


Differences:

- ✓ First activity starts from "zero"
- ✓ EF = ES + Duration
- ✓ ES (Successor) = EF (Predecessor)
- ✓ LF (Predecessor) = LS(Successor)

Similarities:

- ✓ Project duration
- ✓ EF & LF of any activity
- ✓ Float in any activity



Differences:

- ✓ First activity starts from "One"
- ✓ EF = ES + Duration -1
- ✓ ES (Successor) = EF (Predecessor) +1
- ✓ LF (Predecessor) = LS(Successor) -1

Similarities:

- ✓ Project duration
- ✓ EF & LF of any activity
- ✓ Float in any activity



Tools & Techniques: Critical Path Method

- ✓ the minimum amount of time needed for a project to be completed.
- √ the longest continuous path through a network diagram of interdependent tasks
- ✓ delay in the tasks on the critical path will directly impact the overall project timeline.
- ✓ Tasks on the critical path have zero slack or float, meaning there is no flexibility in their scheduling without affecting the project's completion date.
- ✓ Identifying and managing the critical path is essential for project managers to ensure timely project delivery and efficient resource allocation.



Activity	Preceding Activity	Estimate in Months
Start		0
D	Start	4
Α	Start	6
F	D, A	7
E	D	8
G	F, E	5
В	F	5
Н	G	7
С	Н	8
End	C, B	0

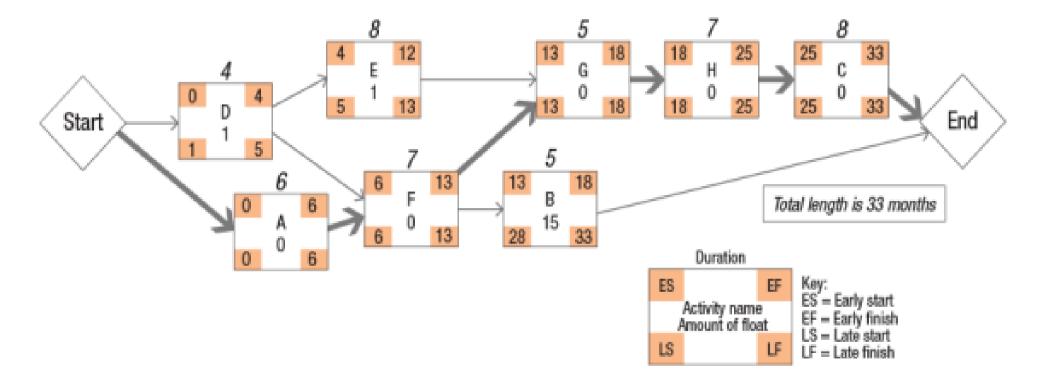
- 1. What is the duration of the critical path in months?
- 2. What is the float of activity B?
- 3. What is the float of activity E?
- 4. What is the float of activity D?

Path	Duration
Start, D, E, G, H, C, End	32
Start, D, F, G, H, C, End	31
Start, D, F, B, End	16
Start, A, F, G, H, C, End	33
Start, A, F, B, End	18



Activity	Preceding Activity	Estimate in Months
Start		0
D	Start	4
Α	Start	6
F	D, A	7
E	D	8
G	F, E	5
В	F	5
Н	G	7
С	Н	8
End	C, B	0

- 1. What is the duration of the critical path in months?
- 2. What is the float of activity B?
- 3. What is the float of activity E?
- 4. What is the float of activity D?





Activity	Preceding Activity	Estimate in Months
Start		0
D	Start	4
Α	Start	6
F	D, A	7
E	D	8
G	F, E D	5
В	F	5
Н	G	7
С	Н	8
End	C, B	0

1. To shorten the length of the project, the sponsor has offered to remove the work of activity E from the project, making activity D the predecessor to activities G and F. What will be the effect?

Path	
Start, D, G, H, C, End	24
Start, D, F, G, H, C, End	31
Start, D, F, B, End	16
Start, A, F, G, H, C, End	33
Start, A, F, B, End	18



Tools & Techniques: Resource Optimization

Resource Optimization is used to adjust the start and finish dates of activities to adjust planned resource use to be equal to or less than resource availability Resource leveling.

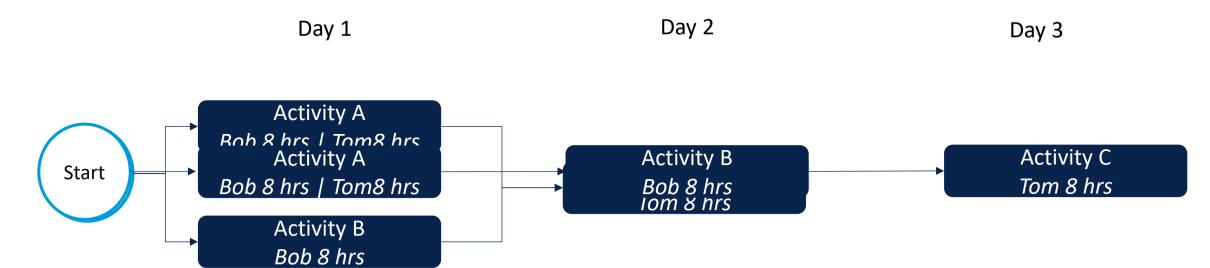
- A technique in which start, and finish dates are adjusted based on resource constraints to balance the demand for resources with the available supply.
- Resource leveling can often cause the original critical path to change. Available float is used for leveling resources. Consequently, the critical path through the project schedule may change.

Resource smoothing.

- A technique that adjusts the activities of a schedule model such that the requirements for resources on the project do not exceed certain predefined resource limits.
- In resource smoothing, as opposed to resource leveling, the project's critical path is not changed, and the completion date may not be delayed.
- In other words, activities may only be delayed within their free and total float. Resource smoothing may not be able to optimize all resources.

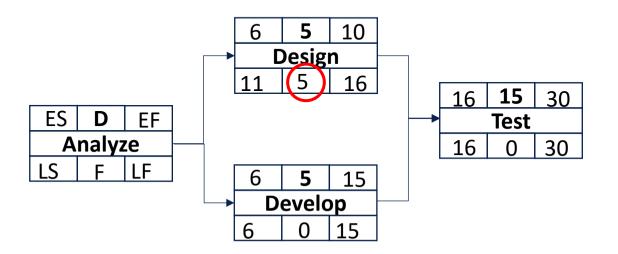
RESOURCE LEVELING





RESOURCE SMOOTHING

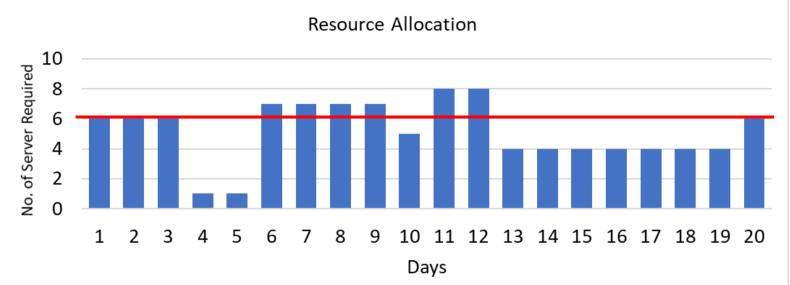






Tools & Techniques: Resource Optimization

Activity		No. of Server Required	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Α	3	6																							
В	2	1																							
С	5	5																							
Е	9	4																							
G	1	6																							
D	4	2																							
F	2	4																							



Available No. of Server



Tools & Techniques: Resource Optimization

			-																						
Activity	Duration (days)	No. of Server Required	1	2	3	4	5	6	7	00	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Α	3	6																							
В	2	1																							
С	5	5																							
Е	9	4																							
G	1	6																							
D	4	2	·									·											·		
F	2	4									·														



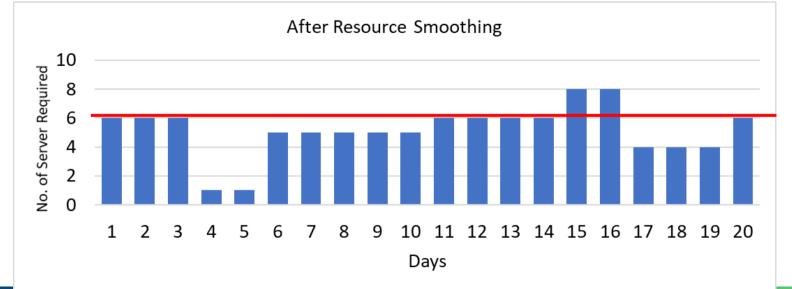
Resource Leveling

- •The downside of resource leveling is that it might extend the overall project duration, as adjustments are made to avoid resource conflicts.
- •Resource leveling may impact the critical path of the project



Tools & Techniques: Resource Optimization

Activity	(days)	No. of Server Required	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Α	3	6																						
В	2	1																						
С	5	5																						
Е	9	4																						
G	1	6																						
D	4	2																						
F	2	4																						



QUIZ TIME (UNGRADED)



According to the draft project schedule, coding work on an IT project spanned for 2.5 months with a 0.5-month free float. The number of developers required for month 1, 2 and 3 were 8, 2, and 5 respectively. According to the budget, the project can afford a maximum of 6 developers in any month. Alice, the project manager, requested Bob, the Development Lead, to adjust the resource requirements for the coding activities to align with the available budget. Melissa rearranged the activities such that the developers required for month 1, 2 and 3 changed to 6, 4, and 5 respectively, and it resulted in extending the coding activity duration to 2.75 months.

This exercise of adjusting the schedule to meet the project constraints is an example of



Tools & Techniques: Resource Optimization

Resource leveling is used when:

- 1. A critical resource may not be available for a certain duration;
- 2. A critical resource may not be available at a certain point of time;
- 3. You have to share a resource with another project;
- 4. The demand for a resource exceeds the supply.

Resource Smoothing is used when:

1. The end date is the main constraint of your project.



Tools & Techniques: Data Analysis

What-if scenario analysis

- What-if scenario analysis is the process of evaluating scenarios in order to predict their effect, positive or negative, on project objectives.
- This is an analysis of the question, "What if the situation represented by scenario X happens?"
- The outcome of the what-if scenario analysis can be used to assess the feasibility of the project schedule under different conditions, and in preparing schedule reserves and response plans to address the impact of unexpected situations.

Simulation

- Simulation models the combined effects of individual project risks and other sources of uncertainty to evaluate their potential impact on achieving project objectives.
- The most common simulation technique is Monte Carlo analysis, in which risks, and other sources of uncertainty are used to calculate possible schedule outcomes for the total project.



Tools & Techniques: Data Analysis

What-if scenario analysis

- What-if analysis is used to explore and compare various plan and schedule alternatives based on changing conditions..
- This is an analysis of the question, "What if the situation represented by scenario X happens?"
- The outcome of the what-if scenario analysis can be used to assess the feasibility of the project schedule under different conditions, and in preparing schedule reserves and response plans to address the impact of unexpected situations.

Some examples might be:

- ✓ What if lead time for major equipment or components is extended?
- ✓ What if we need to extend the duration of certain designing activities?
- ✓ What is the effect on completion date and resources if the current performance trend continues?
- ✓ What if you hire more people?
- ✓ What if your priorities change?



Tools & Techniques: Data Analysis

Simulation

- simulation is a powerful technique to assess schedule risk in project management.
- The most common simulation technique is Monte Carlo analysis.
- Monte Carlo simulation is a method of generating random samples from a specified distribution and using them to simulate the outcomes of a complex system or process.
- In project management, Monte Carlo simulation can be used to model the duration and cost of a project based on the estimates and assumptions of the project team.



Tools & Techniques: Data Analysis

- \checkmark In a **Monte Carlo analysis**, we run the same model selecting a random value for each task
- ✓ We do it hundreds or thousands of times.
- ✓ Each time it runs, we record the values.

Consider a simple project plan. This plan includes a single project, with three tasks:

Task	Best-Case	Expected	Worst-Case
Task 1	10 Days	20 Days	30 Days
Task 2	10 Days	20 Days	30 Days
Task 3	5 Days	10 Days	20 Days
Total	25 Days	50 Days	80 Days

Theit's sause for bhut the ser tead lest in ubserdances it's round like by the steep and notes know idential gestanch task bie steep as estimates on thee task arbeitance it in low. Whomse wilde by some gentaty at task teoloographe to the others finish ahead of schedule.



Tools & Techniques: Data Analysis

✓ Now to add some randomness we will introduce PERT (Program Evaluation Review Technique)

$$PERT = (O + 4M + P) / 6$$

Task	Best-Case	Expected	Worst-Case	Sample (PERT)
Task 1	10 Days	20 Days	30 Days	24 Days
Task 2	10 Days	20 Days	30 Days	16 Days
Task 3	5 Days	10 Days	20 Days	13 Days
Total	25 Days	50 Days	80 Days	53 Days

In this case, with random values selected for each task, you can see that the first task takes a little longer than expected. The last task takes longer as well. The second task is completed ahead of schedule.

However, A single random case is not useful, because it represents only one out of essentially limitless possibilities. This is where we can use Monte Carlo simulation to measure and understand the risk.



Tools & Techniques: Data Analysis

- ✓ In a Monte Carlo analysis, we run the same model selecting a random value for each task but we do it hundreds or thousands of times.
- ✓ Each time it runs, we record the values.
- ✓ When the simulation is complete, we can look at statistics from the simulation' to understand the risk in the model.
- ✓ We'll start with the basic statistics: the minimum, maximum, and average.

Task	Best-Case	Expected	Worst-Case	Sample (PERT)	Minimum	Average	Maximum
Task 1	10 Days	20 Days	30 Days		11 Days	20 Days	29 Days
Task 2	10 Days	20 Days	30 Days		10 Days	20 Days	29 Days
Task 3	5 Days	10 Days	20 Days		5 Days	11 Days	19 Days
Total	25 Days	50 Days	80 Days		32 Days	51 Days	70 Days

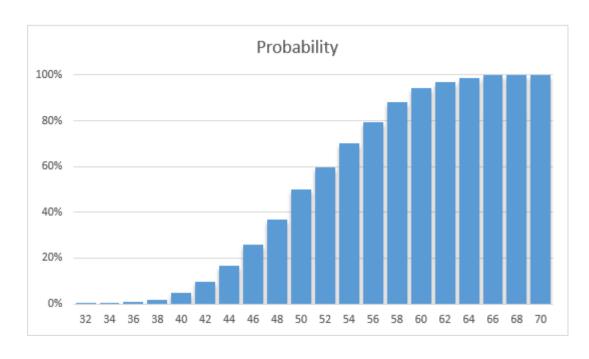
The first thing we can observe that the average completion time is close to our central estimate. That is what we would expect; it's the most likely case for each task. We can now say that the worst case scenario is 70 days, instead of 80.

This data comes from a Monte Carlo analysis of 1,000 trials.

Next, we can look at the range of data generated during the Monte Carlo simulation to understand more about the projections.



Tools & Techniques: Data Analysis



- ✓ there is only a 5% chance that the project will be completed within 40 days
- ✓ there is a 94% chance the project will be completed within 60 days.

What does it mean?

- In only 5% of the trials (50 out of 1,000) the total project time was 40 days or less. We can therefore say that, during the simulation, there was a 5% probability that the project would be completed within 40 days.
- Similarly, during the simulation, there was a 94% chance that the project was completed within 60 days.

Key Points

- ✓ Estimated Range is 32-70 Days (Even though our best- and worst-case estimates result in totals of 25-80 days)
- ✓ Probability of Completion within 50 Days is 50%
- ✓ For 75% Confidence, Budget 55 Days; for 85% Confidence, Budget 58 Days
- ✓ We can say that in 75% of the simulation trials, the project was completed within 55 days. We can budget for various confidence levels by looking at probability from the simulation.



Tools & Techniques:

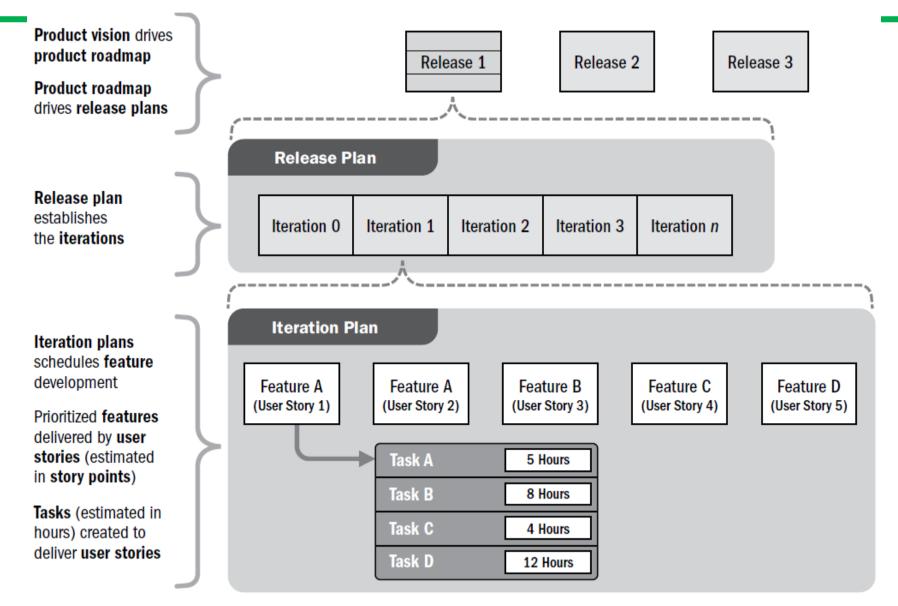
Leads and lags

Project Management Information System (PMIS)

Agile Release Planning

- ✓ Agile release planning provides a high-level summary timeline of the release schedule (typically 3 to 6 months) based on the product roadmap and the product vision for the product's evolution.
- ✓ also determines the number of iterations or sprints in the release and allows the product owner and team to decide how much needs to be developed and how long it will take to have a releasable product based on business goals, dependencies, and impediments





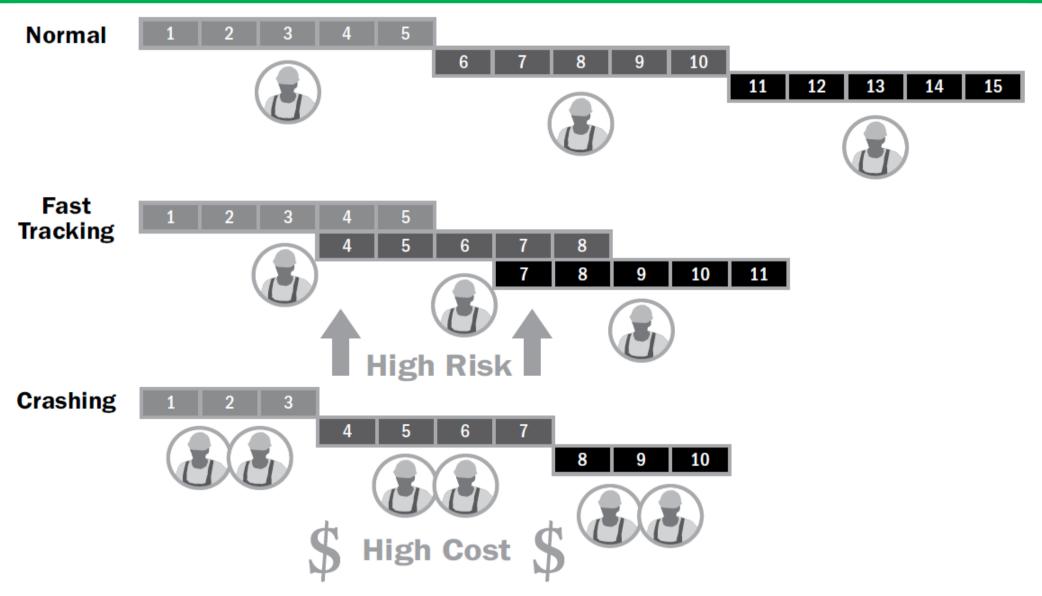
Relationship Between Product Vision, Release Planning, and Iteration Planning



Tools & Techniques: Schedule compression

- ✓ Schedule compression techniques are:
 - Crashing.
 - > With crashing, resources are added to the activities on the critical path, and the duration of the activities is reduced as a result.
 - Fast tracking.
 - A schedule compression technique in which activities or phases normally done in sequence are performed in parallel for at least a portion of their duration





Schedule Compression Comparison



Tools & Techniques: Schedule compression - Crashing

Time Is Money: Cost-Time Tradeoffs

✓ Reducing the time of a critical activity usually incurs additional direct costs.

Reasons for imposed project duration dates:

- ✓ Time-to-market pressures
- ✓ An unrealistic schedule
- ✓ Unforeseen delays
- ✓ Incentive contracts (bonuses for early completion)
- ✓ Imposed deadlines and contract commitments
- ✓ Overhead and public goodwill costs
- ✓ Pressure to move resources to other projects



Tools & Techniques: Schedule compression - Crashing

Determining Activities to Shorten

Shorten the activities with the smallest increase in cost per unit of time.

Assumptions:

The cost relationship is linear.

Normal time assumes low-cost, efficient methods to complete the activity.

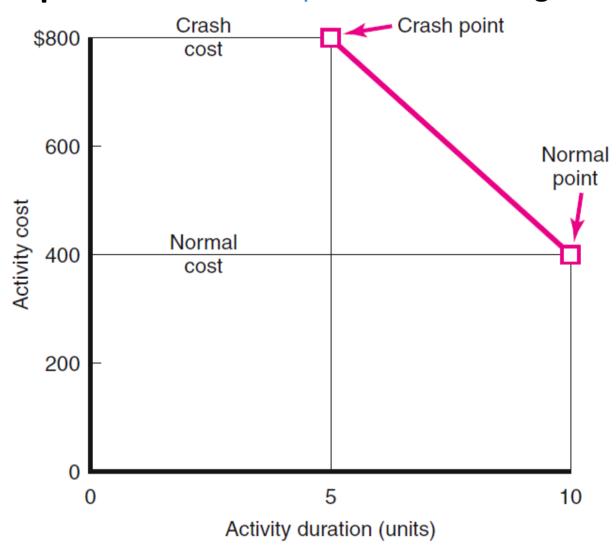
Crash time represents a limit—the greatest time reduction possible under realistic conditions.

Slope represents a constant cost *per unit of time*.

All accelerations must occur within the normal and crash times.



Tools & Techniques: Schedule compression - Crashing





Tools & Techniques: Schedule compression - Crashing

Based on the data provided in Table1, calculate the lowest cost of crashing to complete the project four weeks earlier than the baseline schedule. What activities should go through crashing for that?

Activity	Predecessor	Baseline		Crash		
		Time	Cost (\$)	Time	Cost (\$)	
Α		10	4000	8	6000	
В	Α	2	6000	2	6000	
С	Α	14	22000	12	26000	
D	Α	6	9000	5	10000	
E	B,D	9	14000	7	19000	
F	C,E	8	18000	4	36000	



Tools & Techniques: Schedule compression - Crashing

Based on the data provided in Table1, calculate the lowest cost of crashing to complete the project four weeks earlier than the baseline schedule. What activities should go

through crashing for that?

✓ Identify the critical path in the baseline schedule (A-B-C-F).

•Activity A: Baseline Time Savings = 10–8=2 weeks

•Activity C: Baseline Time Savings = 14–12=2 weeks

•Activity F: Baseline Time Savings = 8–4=4 weeks

Activity	Predecessor	Bas	eline	Crash		
		Time	Cost (\$)	Time	Cost (\$)	
A		10	4000	8	6000	
В	A	2	6000	2	6000	
С	A	14	22000	12	26000	
D	A	6	9000	5	10000	
E	B,D	9	14000	7	19000	
F	C,E	8	18000	4	36000	

Now, let's calculate the cost:

•Total Cost of Crashing:

• Total Cost of Crashing=Sum of (Additional Crash Time Needed × Crash Cost Per Time Saved)

Activity A:

Crash Cost Per Time Saved = (6000-4000)/2=1000

Total Cost of Crashing for $A = 2 \times 1000 = 2000$ dollars

Total Cost of Crashing for $C = 2 \times 2000 = 4000$ dollars

Total Cost of Crashing for $F = 4 \times 4500 = 18000$

Total Cost of Crashing = 2000 + 4000 = 6000



Output:

Schedule Baseline

- Specific approved version of the project schedule model
- Can be changed through formal change control
- Used as the basis for comparison

Project Schedule

- Bar chart
- Milestone Chart
- Project schedule network diagram

Schedule Data

- Resource requirement by the time period,
- Alternative schedule (best case, worst case),
- Scheduling of contingency reserve

Project calendars

- Identifies working days and shifts
- Time period in days or part in days is available or not available



Output:

Change Requests • Modifications to the project scope or project schedule may result in change requests to the scope baseline, and/or other components of the project management plan.

Project management plan update

- Schedule management plan
- Cost baseline

Project
Document
update

• Activity attribute, Assumption log, Duration estimates, LL register, Resource requirement, Risk register



Data Date

Detailed Schedule

Activity	Activity Description	Calendar units	Project Schedule Time Frame					
Identifier			Period 1	Period 2	Period 3	Period 4	Period 5	
1.1.MB	Begin New Product Z	0						
1.1	Develop and Deliver Product Z	120						
1.1.1	Work Package 1: Component 1	67						
1.1.1.D	Design Component 1	20		FS	1			
1.1.1.B	Build Component 1	33		-	╞┧			
1.1.1.T	Test Component 1	14	ss	G				
1.1.1.M1	Complete Component 1	0	33					
1.1.2	Work Package 2: Component 2	53			 			
1.1.2.D	Design Component 2	14		<u> </u>				
1.1.2.B	Build Component 2	28	-		Data Date			