DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

LAB MANUAL – SOFTWARE PROJECT MANAGEMENT

B. Tech – III/VI

REGULATION – 2022

INDEX

SOFTWARE ARCHITECTURE AND DESIGN PATTERNS LAB

EX. NO: 1. SETTING UP PROJECT MANAGEMENT TOOLS (E.G., JIRA, TRELLO) AND INTRODUCING A SAMPLE PROJECT.

7 steps to get started in Jira

Jira is a work management tool for software teams that need to organize and track their work. Jira is incredibly flexible and can be customized to work with your team's unique workflow, meaning teams of all kinds can enjoy increased productivity and visibility as they march toward releasing amazing software.
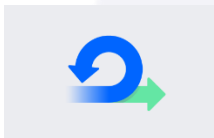
Step 1 - Create a project

Log into your Jira site. In the top navigation, select the "Projects" dropdown and select "Create project".

Step 2 - Pick a template

There are dozens of Jira templates, each of which is designed to get your team started quickly and successfully. Today, there are three templates specifically for software teams:

Scrum

For agile teams that work from a backlog, plan and estimate their work in sprints, and deliver work on a regular schedule.

Kanban

For agile teams that monitor work in a continuous flow (rather than in sprints), with a focus on managing in-progress work. (Includes the option of a kanban backlog.)

Bug tracking

For teams that don't need boards and prefer to manage development tasks and bugs in a list view.

Project Types

For the **scrum and kanban templates only**, you will also be prompted to choose a project type. The fundamental difference between the two project types is how they are administered, and whether that occurs at the team level or at a company/Jira admin level.

Team-managed projects are suited for independent teams who want to control their own working processes and practices in a self-contained space.

Company-managed projects are set up and maintained by Jira admins. This project type is designed for teams who want to standardize a way of working across many teams, such as sharing a workflow.

Step 3 - Set up your columns

In Jira, the board displays a selection of issues in columns, with each column representing a step in your team's workflow for taking work through completion. Although there are many things you can configure on your board, we suggest just setting up columns for now. When you're getting started on a new Jira project, it's important to make your board reflect the way your team works.

How you setup the columns on your board in the scrum and kanban templates depends if you're in a team-managed (directly from the board) or company-managed project (in board settings).

In team-managed projects:

- Go to your board. Select (•••) in the top right and click Configure board.
- Add a new column, change the name of column, delete a column, or a move a column as necessary.

In company-managed projects:

- Go to your board. Select (•••) in the top
- right and click Board settings.
- Click the Columns tab.
- Add a new column, change the name of column, delete a column, or a move a column as necessary.
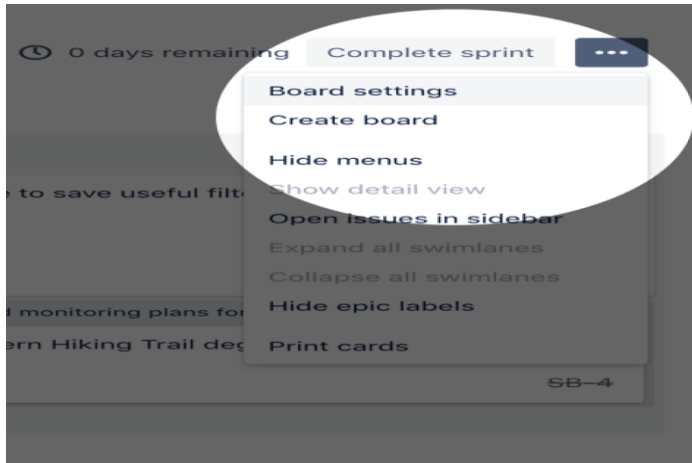
Step 4 - Create an issue

Issues are the building blocks of your Jira project. An issue can represent a story, epic, bug, feature to be built, or any other task in your project.

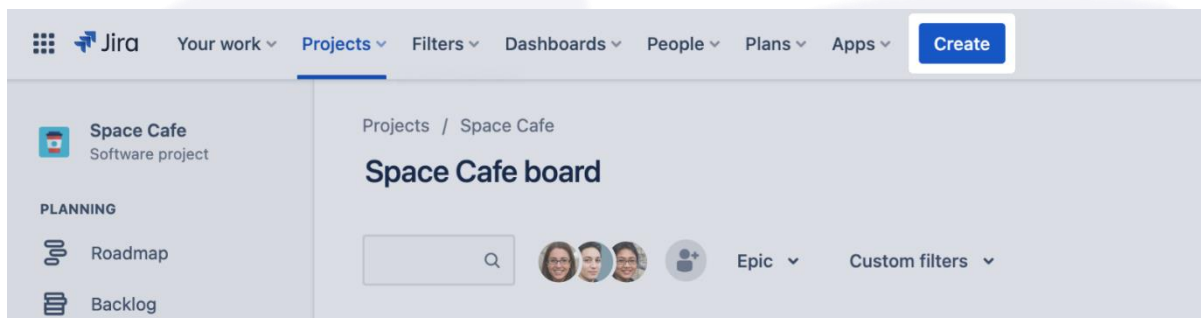Select "**Create**" in the top navigation. Your issue will appear in the backlog or board of the project.

Step 5 - Connect your tools

You and your teams can spend less time managing work and more time building great software with over 3,000 Jira apps that can be custom-tailored to fit any and every use case. Build a unified workspace with 3000+ apps and integrations from the Atlassian Marketplace.

1. Select the **Cog** in the right corner of the top navigation > **Apps**.

2. Click **Find new apps**.

3. Search by the app name, or choose a category.

4. Follow the prompts to **Install**, **Buy now**, or start a **Free trial**.

Step 6 - Invite your team

It's time to get the party started! Once you have enough work represented on your board, start inviting team members.
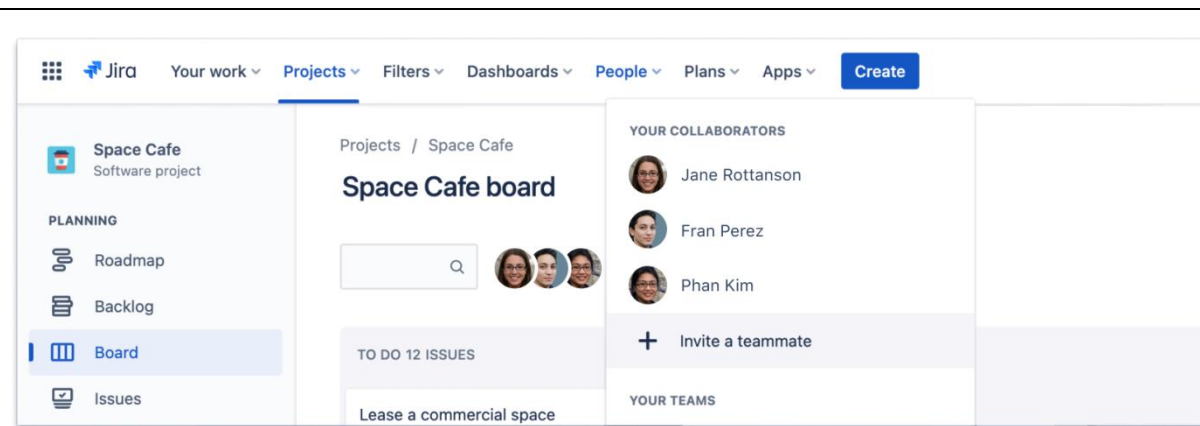
Step 7 - Move work forward

Now that your team has joined your Jira site, you're ready to collaborate and track work together. If you're in a scrum project, you'll need to create and start a sprint to begin tracking work. If you're in a kanban project, you can start tracking work on the board. To track work items, move an issue from one column to another as it progresses through your team's workflow.
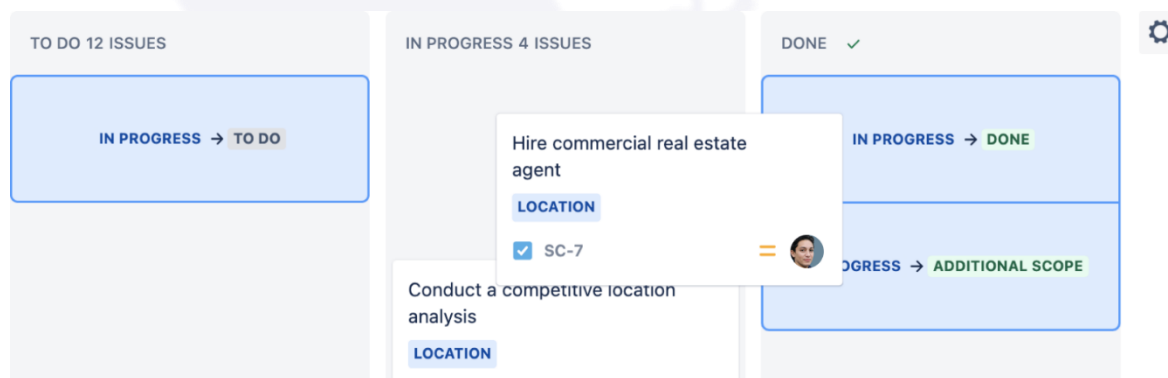
Result:

By setting up a project management tool like Jira or Trello, you can:

- Effectively plan and manage tasks.

- Assign clear roles and responsibilities.

- Monitor project progress in real-time.

- Collaborate seamlessly with your team.

EX.NO: 2. CREATING A PROJECT PLAN WITH MILESTONES, TASKS, AND RESOURCE ALLOCATION.

To Make a Project Plan in 4 Steps

A **project plan** contains the schedule, tasks, roles and other key information of a professional project.

Project planning refers to the phase in project management in which you determine the actual steps to complete a project. This includes laying out timelines, establishing the budget, setting milestones, assessing risks, and solidifying tasks and assigning them to team members.

Project planning is the second stage of the project management lifecycle. The full cycle includes initiation, planning, execution and closing.

A **project plan** is a document that lays out the key information of a project. This can vary depending on the organisation and project. The components of a project plan typically clarify:

- **Scope and goals:** A project plan should make clear what the project is aiming to achieve.

- **Schedule:** The schedule outlines when the project will start and end, how long tasks are expected to take and when milestones should be reached.

- **Tasks and milestones:** Tasks are the components of work that must be completed to achieve milestones and eventually the entire project. Milestones are a set of tasks that define the end of a phase of the project. For example, completing a website prototype in a project to redesign a company's website would be considered a milestone.

- **People:** A project plan generally defines which individual is in charge of what task.

- **Documentation:** A project plan might include links to other important charts and documents, like RACI charts, a project charter, budget, or risk management plan, so that it is easy to find key information.

Project plan template

A template can provide project managers with a starting point that they can customise to their needs. Many are available for free download online like this project plan template, from the Google Project Management: Professional Certificate, which uses Google sheets. Other templates use Microsoft Word, Google Docs, or Microsoft Excel.

Microsoft Excel Gantt chart template from Microsoft

Project Planner.

Select a period to highlight at right. A legend describing the charting follows.
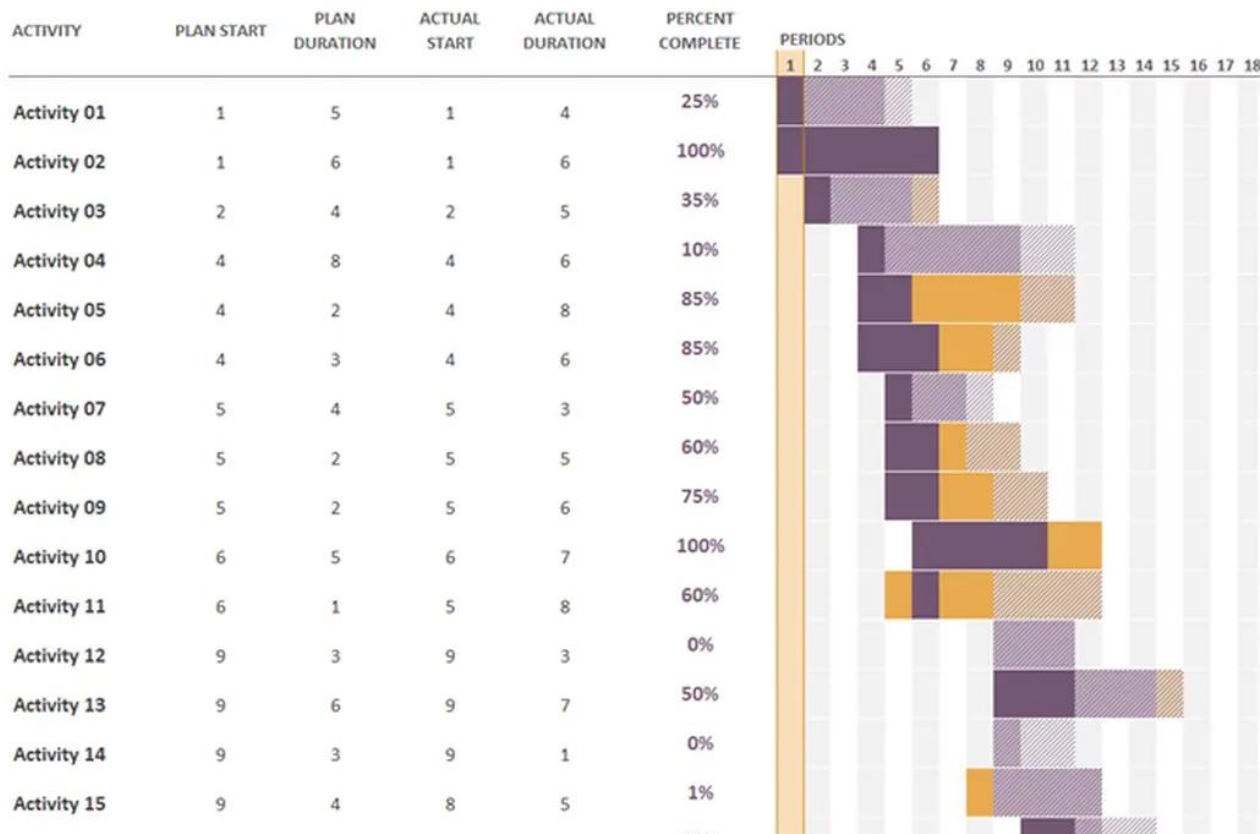
Period Highlight: 1 ▨ Plan Duration ▨ Actual Start ▨ % Complete

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Activity 01	1	5	1	4	25%
Activity 02	1	6	1	6	100%
Activity 03	2	4	2	5	35%
Activity 04	4	8	4	6	10%
Activity 05	4	2	4	8	85%
Activity 06	4	3	4	6	85%
Activity 07	5	4	5	3	50%
Activity 08	5	2	5	5	60%
Activity 09	5	2	5	6	75%
Activity 10	6	5	6	7	100%
Activity 11	6	1	5	8	60%
Activity 12	9	3	9	3	0%
Activity 13	9	6	9	7	50%
Activity 14	9	3	9	1	0%
Activity 15	9	4	8	5	1%

PERIODS 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

How to create a project plan

Your exact project plan might look different depending on the preferences of the project manager and the organisation. Generally, however, you can start with determining your timeline before going on to solidify tasks, milestones, and roles and compiling other important documents.

1. Determine a timeline.

The cornerstone of the project plan is often the timeline or schedule. A timeline should include the date you will begin and expect to end the project, how long it will take to finish each task and milestone, and the dates you expect tasks and milestones to be completed.

Project managers often begin creating schedules around hard constraints determined by stakeholders. Do you need to design and produce a new toy before the holiday shopping season? You will want to make sure your schedule reflects this. Be sure to speak with team members to get a sense of how long each task typically takes. You may also want to include time buffers for tasks that involve some risk.

Tools at this stage you can use include:

- Gantt chart

- Work breakdown structure

2. Build out tasks and milestones.

Once you know when tasks, milestones, and the whole project should be completed, you can determine what resources are needed at what point in the project and which of your team members will work on each task. This exercise is called capacity planning.

You can also use this time to determine the critical path in a project. The critical path is the bare minimum of tasks you must complete to meet the project goal.

3. Establish roles.

In this phase, solidify the tasks each team member is assigned, and communicate with them to make sure they are informed and have their questions answered.

If you have created a RACI chart in the project initiation phase, this will be a good time to refer to it.

Link to important documents.

A project plan often becomes a central document referred to often as the project progresses. It might be a good idea to attach or link documents that will be useful to have on hand. If your project plan is in a spreadsheet, you might link to other documents in separate tabs for easy access.

Important documents might include:

- Project charter

- Project budget

- Communication plan

- RACI chart

- Risk management plan

- Change management plan

Popular project planning tools include:

- Monday.com Team Gannt

- These tools allow you to visually represent a project timeline, assign tasks to team members, and more. You can start a project plan for free on both (as of time of writing).

EX. NO: 3. CONDUCTING A PROJECT KICKOFF MEETING AND DEFINING PROJECT SCOPE.

A project kick-off meeting is the first meeting with the project team and the client of the project where applicable. This meeting is the time to establish common goals and the purpose of the project.

https://www.atlassian.com/software/confluence/templates/project-kickoff

Starting a project without a kick-off meeting is like setting off on a trip without any concrete plan. You're likely to see some cool things along the way, but you'll probably end your vacation with

some weary travellers and a fairly sparse photo album. A good project kick-off meeting sets the course for a successful and smooth project collaboration.

Why should we have a project kickoff?

As the first meeting between the project team members and possibly the client or sponsor, the project kick-off is the best time to set expectations and foster strong team morale. Usually, the kick-off takes place after the statement of work or project poster has been finalized and all parties are ready to go.

our kick-off is an opportunity to orient the team to the work at hand, decide how everyone will work together, and establish common project goals and check-ins. Think about discussing things like how you'll communicate, how often you'll meet, what the timeline is, and what could slow the project down (and how to avoid that).

Planning your project kick-off meeting

Pre-work

A project kick-off meeting shouldn't be an "information broadcast." If you need to share background information in advance, do it on a shared document like a Confluence page. Project kick-off meetings should actively involve the team and anyone else who's a stakeholder or whose work will be affected by the project.

Set up a meeting agenda to keep the meeting streamlined and efficient, try making a list of questions you want to ask your team. (Even better, send those questions to them ahead of the meeting so they'll have time to think about their answers.) Prepare some answers based on questions you think your team might have.

Structuring the meeting

How you structure your meeting agenda depends on the project, but key elements should include the 5 W's, or *who*, *what*, *where*, *when*, *why* and *how*:

- Introductions
- What's the background of the project
- Why are you doing it
- What is the project scope
- What's the action plan
- Who's doing what
- How are you going to work together
- What does success look like

What's involved in a kick-off meeting?

Statement of work, project scope, timeline, and deliverables

Out of everything you're discussing during the project kick-off meeting, this is one of the most important parts. It solidifies what's involved in the project and whether or not everyone is on the same page.

The statement of work you submitted to the client describes the work you'll deliver and by what deadline.

The project scope includes a detailed description of project deliverables and the work needed to accomplish them.

The project timeline is an overview of what is expected to be delivered and when. This can be as detailed as needed for your project.

Deliverables are the actual items you will send to your customer or client, and they're usually noted on the timeline.

Tracking progress/milestones and communicating them

During your kick-off meeting, establish a baseline for how you plan to share progress with your stakeholders or client. You can use a status report to show how the project is tracking on budget, tasks, milestones, and timeline. The status report should be available to all stakeholders and anyone whose work might be affected by the project, to make sure everyone is aware of the progress (or delays). For client projects, it's important that your client agrees with the level of detail they will receive in the status reports to ensure easier invoicing and payments.

Tools and methods

Put together a set of collaboration tools you'll be using during the project and decide with your team how you'll be using them to communicate. How will you be sharing your status reports and other files? Aligning on these things early on will help to keep the project's communication streamlined. It's also helpful a to make sure your team or client can access whatever tools you decide to use to keep informed about the project. Some project tools you may use include:

- A work management system to track items in progress. Examples include Jira, Basecamp, or Asana.
- A shared document system to make sure everyone (including your client) has access to important project documents and information. Examples include Microsoft docs, Google

docs, or you can use a document management system that has built-in hierarchy and organization tools like Confluence.

- A communication tool like email or Slack so that the teams can stay in touch about any questions or needs that emerge.

For client work, there may be other tools you'll need access to before you can started. Think design assets like logos, fonts, or style guides, templates, databases, logins (to CMS, social, analytics, etc), access to your client's intranet, and the correct details for invoicing.

EX.NO: 4. DEVELOPING A WORK BREAKDOWN STRUCTURE (WBS) FOR A GIVEN PROJECT.

The output of the WBS development process might seem simple: a short document with a list of deliverables. To create it, however, you need a thorough understanding of the project's scope, your team's capabilities, and your stakeholders' requirements.

Here's a process for creating a WBS from scratch.

1. **Understand the Project's Scope**

   In our earlier project management guide, we identified the WBS as one of the key documents created at the end of the 'Planning' phase'.
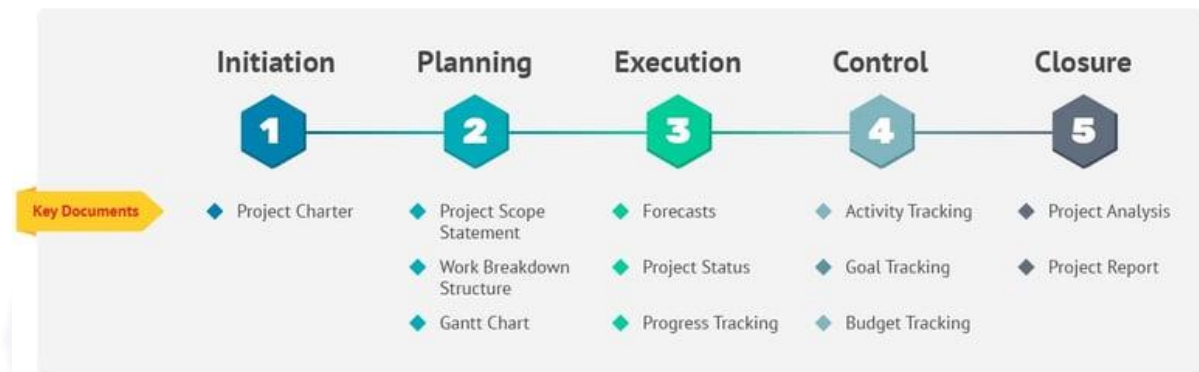
   However, before you can create it, you need a thorough understanding of the project's scope and objectives.

Chiefly, you need two things:

- Project scope statement to understand the project's scope in detail.
- Project scope management plan to understand how to deal with changes to the project's scope (which will affect your deliverables).

   You'll want to refer to your project charter to develop the scope statement and scope management plan.

   The output of the entire WBS development process is as follows:

- Work breakdown structure
- WBS dictionary
- Scope baseline

2. **Determine Major Deliverables**

Once you understand the project scope, start the WBS development process by determining the key deliverables.

For example, if your goal is to "build a house", you might have the following three broad deliverables at Level 2:
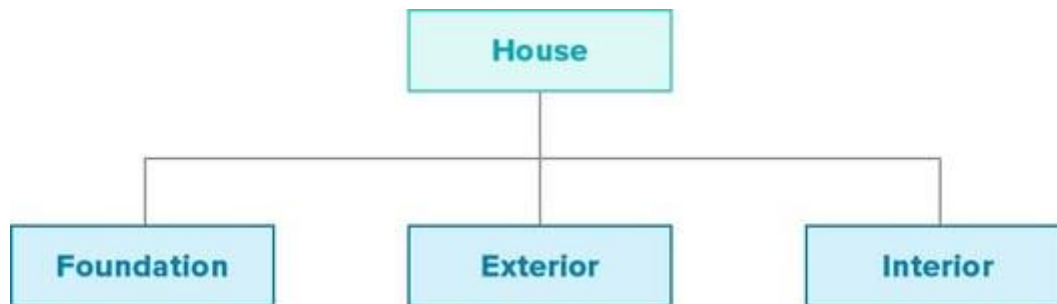
There are two heuristics you can follow for determining major deliverables at the 2nd level:

- Each deliverable must be essential to the project's success. For example, you can't build a house without a foundation, exterior or interior.
- Each deliverable should be the responsibility of an independent team. In the above example, the team responsible for laying the foundation won't be like the team building the interiors.

3. **Determine Work Packages**

As you learned above, a work package is a deliverable at the lowest level of a WBS.

In a typical 3-level WBS, determining work packages would be the next step after identifying major deliverables.

This is one of the most important parts of the WBS development process, and it will require extensive input from your project team and stakeholders.

You aim to pick a major deliverable and identify all the work necessary to complete it. This work package must be:

- Independent: The work package must be mutually exclusive and not depend on other ongoing elements.
- Definable: The work package should have a definite beginning and end and should be understood by all project participants.
- Estimable: You should be able to estimate the work package's duration and resource requirements.

- Manageable: The package must represent a "meaningful unit of work," i.e., it must accomplish something concrete and can be assigned to an individual or team. It should also be measurable.
- Integratable: The package must integrate with other elements to create the parent level.
- Adaptable: Ideally, the package can accommodate changes in scope according to the project's requirements.

If the work can't meet the above requirements, you can decompose the WBS into another level.

There are a few heuristics you can follow for determining work packages:

- **8/80 rule:** A common rule of thumb is that each work package must be no longer than 80 hours and no less than 8 hours in total length. If it is longer, decompose it further. If it is shorter, think of going up by one level.
- **Reporting period:** Another common rule is to limit each work package to a single reporting period. If it takes longer than one reporting period (monthly, weekly, etc.) to accomplish, decompose it further.
- **Use nouns:** You should be able to describe each work package with a noun or an adjective. To break it down further, you'll need to use verbs. For example, "bike seat" describes a work package. If you break it down further, you must use verbs like "cut foam", "stitch leather", etc.

4. Create a WBS Dictionary

The WBS dictionary outlines the definition and scope of each element in the WBS. It is a supporting document meant to help incoming project teams better understand each work package.

You don't necessarily need a WBS dictionary, especially if the project is simple or has a limited scope. However, the dictionary can greatly improve clarity for complex projects with a lot of churn.

Further, the WBS dictionary takes you one step closer to creating the project schedule. You can often transplant details from this dictionary straight to your project scheduling tool.

Here are a few details you can include for each item in the WBS dictionary:

- Work package ID (see the ID convention below)
- Work package name
- Work package description
- Assigned to (individual or team name)
- Department
- Date of assignment

- Due date
- Estimated cost

Here's a more simplified WBS dictionary example with element ID, name, and description:

WBS Level	WBS Code	WBS Name	WBS Description	PWS/SOW Mapping
1	1	Aircraft System	X Series Aircraft System to fly to the moon	
2	1.1	Air Vehicle	X Series Air Vehicle to fly to the moon	
3	1.1.1	Air Frame	X seriese air frame	
3	1.1.2	Avionics	Brains behind the air frame	
4	1.1.2.1	Comms	Communications	
5	1.1.2.1.1	Antenna	Thinging to pick up signals	1.1, 1.2, 1.3.1, 1.13.1
5	1.1.2.1.2	Receiver	Box to interpert signals picked up by Antenna	1.1, 1.2, 1.13.2
5	1.1.2.1.3	Transmitter	Box to send out signals when we talk	1.1, 1.2, 1.3.1, 1.13.3
4	1.1.2.2	Navigation	Back seat driver	
4	1.1.2.3	Fire Control	Off/Def weapons in case we run into aliens	
3	1.1.3	Propulsion	Engine to propel x series air frame	
2	1.2	Training	Training for operation and maintenance of X Series Air Vehicle	

A WBS dictionary helps project team members understand each element.

5. Use the Right WBS Format

Once you have all the work packages and the WBS dictionary, it's time to create the WBS.

There are several WBS formats you can follow. The simplest way to do this is to create text-based hierarchical groupings. By convention, you use numbers and decimal points to indicate the level of the element.

For example, the number 1.1.1.3 means that you're referencing the 3rd element of the 4th level of the WBS.

Thus, you might have a text-based WBS as follows:

```
1.0 Complete System
    1.1 System Component A
        1.1.1 Element #1 of Component A
        1.1.2 Element #2 of Component A
        1.1.3 Element #3 of Component A
    1.2 System Component B
        1.2.1 Element #1 of Component B
        1.2.2 Element #2 of Component B
        1.2.3 Element #3 of Component B
```

Alternatively, you might use a more visual tabular structure as follows:

Level 1	Level 2	Level 3
1 Complete System	1.1 System Component A	1.1.1 Element #1 of Component A
		1.1.2 Element #2 of Component A
	1.2 System Component B	1.2.1 Element #1 of Component B
		1.2.2 Element #2 of Component B
	1.3 System Component C	

Another option is to create a flowchart:
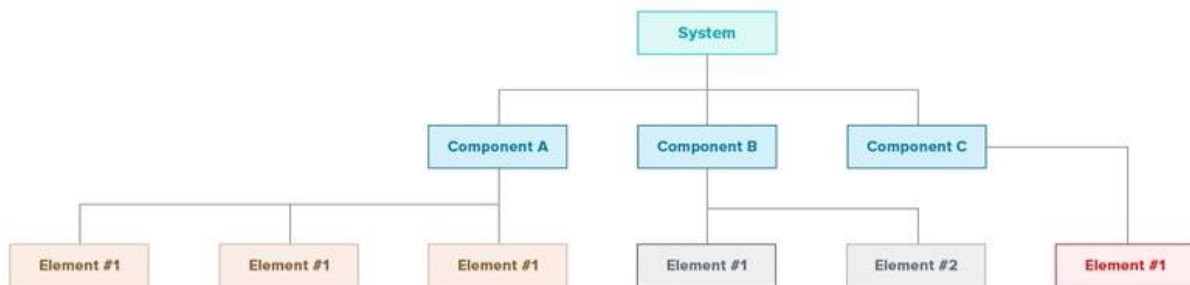
Once you've made the work breakdown structure, share it with your team. Use it to get a high-level overview of the project's progress.

Work Breakdown Structure Template

While creating a work breakdown structure is technically easy (it's just a flowchart or an Excel sheet), it can be time-consuming.

EX. NO: 5 IDENTIFYING AND ANALYZING PROJECT RISKS.

Steps for Identifying and Analyzing Project Risks

Step 1: Risk Identification

The goal is to uncover all potential risks that could affect the project.

1. **Techniques for Risk Identification**:
   - o **Brainstorming**: Collaborate with the team to list all possible risks.
   - o **SWOT Analysis**: Identify strengths, weaknesses, opportunities, and threats.
   - o **Expert Interviews**: Gather insights from stakeholders and subject-matter experts.
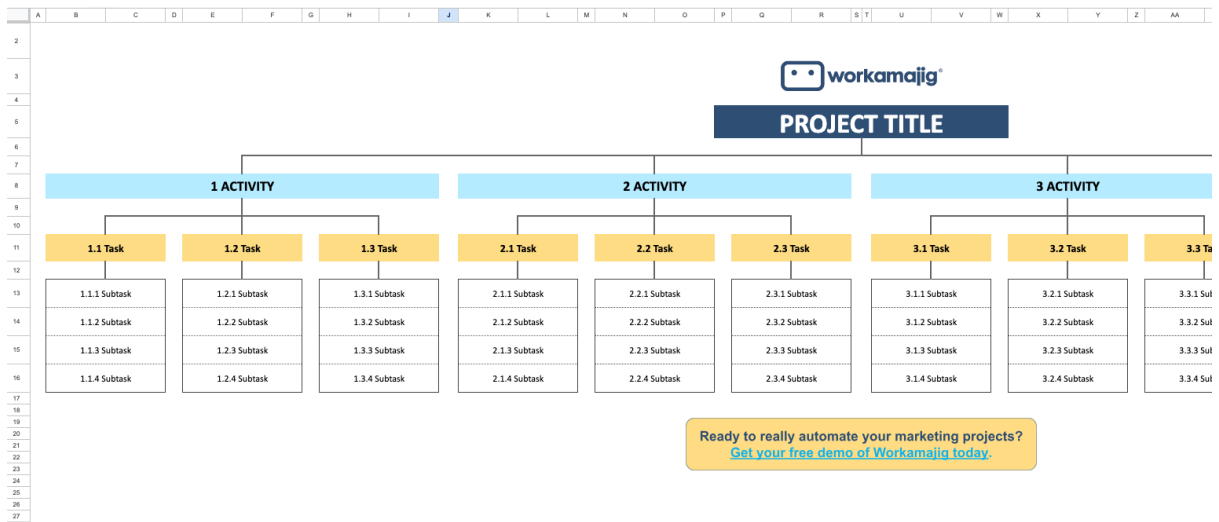   - o **Checklists**: Use predefined risk categories or checklists to identify common risks.
2. **Types of Risks**:
   - o **Technical Risks**: Issues related to technology or tools (e.g., system failures).
   - o **Resource Risks**: Unavailability of skilled personnel or materials.
   - o **Schedule Risks**: Delays in task completion or dependencies.
   - o **Financial Risks**: Budget overruns or unexpected costs.
   - o **External Risks**: Regulatory changes, market conditions, or natural disasters.

Step 2: Document the Risks

Record risks in a **Risk Register** with the following fields:

- Risk ID
- Description
- Category
- Likelihood (Low, Medium, High)
- Impact (Low, Medium, High)
- Responsible Person

Step 3: Risk Analysis

The goal of risk analysis is to prioritize risks based on their potential impact and likelihood.

1. **Qualitative Risk Analysis**:
   - o **Assess Likelihood and Impact**: Use a simple scale (e.g., 1–5 or Low/Medium/High).
   - o **Risk Matrix**:

Likelihood ↓ Impact →	Low	Medium	High
High	Medium	High	Critical
Medium	Low	Medium	High
Low	Low	Low	Medium

   - o **Risk Score**: Multiply likelihood by impact to rank risks.
2. **Quantitative Risk Analysis**:
   - o Use numerical techniques like:

- ▪ **Expected Monetary Value (EMV)**: EMV = Probability × Impact
- ▪ **Monte Carlo Simulation**: Simulates different project scenarios to predict outcomes.

Step 4: Develop a Risk Response Plan

For each identified risk, determine the appropriate response:
- **Avoid**: Change the project plan to eliminate the risk.
- **Mitigate**: Reduce the likelihood or impact of the risk.
- **Transfer**: Shift the risk to a third party (e.g., insurance, outsourcing).
- **Accept**: Acknowledge the risk and prepare a contingency plan.

Example: Identifying and Analyzing Risks

Scenario: Software Development Project

1. **Identified Risks**:
   - o Requirement changes mid-project.
   - o Delays in resource availability.
   - o Server downtime during deployment.
   - o Regulatory compliance issues.

2. **Qualitative Analysis**:

Risk	Likelihood	Impact	Risk Score	Priority
Requirement changes	High	High	15	Critical
Resource delays	Medium	High	10	High
Server downtime	Low	Medium	4	Medium
Compliance issues	Medium	Low	3	Low

3. **Risk Response**:
   - o **Requirement changes**: Schedule regular client reviews to lock requirements.
   - o **Resource delays**: Maintain a resource buffer and cross-train team members.
   - o **Server downtime**: Use a backup server and test deployment in a staging environment.
   - o **Compliance issues**: Consult legal experts early in the project.

Tools for Risk Identification and Analysis

1. **Risk Management Software**:
   - Risk Watch
   - Active Risk Manager
2. **General Tools**:
   - Microsoft Excel: For risk matrices and registers.
   - Jira: For tracking and managing risks in Agile projects.
3. **Templates**:
   - Risk Register templates available on platforms like Smartsheet.

Result:

By systematically identifying and analyzing risks, project managers can proactively address potential challenges and ensure project success. This process fosters transparency, enhances preparedness, and reduces the likelihood of unexpected disruptions.

EX.NO:6 DEVELOPING A QUALITY MANAGEMENT PLAN AND IMPLEMENTING QUALITY CONTROL MEASURES

Aim:

To understand and apply the process of developing a Quality Management Plan (QMP) and implementing Quality Control (QC) measures to ensure project deliverables meet predefined quality standards.

1. **Software Required:**
   o Microsoft Excel or Google Sheets for documenting the QMP.
   o Testing tools (e.g., Selenium, Postman, JUnit) for QC activities.
   o A project management tool (e.g., Jira or Trello) for tracking tasks and defects.
2. **Team Roles:**
   o **Project Manager**: Oversees QMP development.
   o **Quality Analyst**: Executes QC measures.
   o **Team Members**: Perform assigned tasks and testing.

Part A: Developing a Quality Management Plan

Steps

1. **Define Quality Objectives**

   o List the project's quality goals (e.g., defect-free code, performance metrics).

   o Example: "Achieve less than 2 critical defects per release."

2. **Determine Quality Standards**

   o Identify applicable industry standards (e.g., ISO 9001) or project-specific benchmarks.

   o Example: "Follow OWASP security guidelines."

3. **Specify Quality Metrics**

   o Example Metrics:

      ▪ Defect rate = Number of defects / Total code modules.

      ▪ Response time < 2 seconds under standard load.

4. **Establish QA Processes**

   o Define activities to ensure quality:

- ▪ Peer code reviews.

- ▪ Scheduled audits.

- ▪ Risk analysis.

5. **Define QC Procedures**

   o List QC techniques for testing deliverables:

      ▪ Unit Testing: Testing individual functions.

      ▪ System Testing: Verifying complete system functionality.

6. **Assign Roles and Responsibilities**

   o Create a responsibility matrix (e.g., RACI Matrix).

7. **Document the QMP**

   o Use a template to document the QMP.

   o Include sections: Objectives, Standards, Metrics, QA Processes, QC Procedures, Roles, and Continuous Improvement Plans.

(PM) ProjectManager

Quality Control

Issue	Description	Imperfections	Identified By	To Be Fixed By	Priority	Status	Date Opened	Date Closed	Notes
Installation	Not connecting as designed	10	Jeff T.	Sam R.	Critical	Open	5-Apr		
Functionality	Needs to work more seamlessly	3	Jeff T.	John P.	Low	Open	9-Apr		
Stress	Breaking under normal use	5	Jeff T.	John P.	Medium	Open	25-Mar		
Performance	Not meeting performance expectations	9	Margot S.	Sam R.	High	Closed	3-Apr	9-Apr	

Part B: Implementing Quality Control Measures

Steps

1. **Setup Testing Tools**

   o Install and configure a testing tool (e.g., Selenium for automation testing).

2. **Perform Testing**

   o **Unit Testing**: Validate individual modules.

   o **Integration Testing**: Test interactions between components.

   o **System Testing**: Ensure the entire system functions as intended.

3. **Inspect Deliverables**

   o   Use checklists to verify quality standards.

   o   Example Checklist:

      ▪   Code follows coding standards.

      ▪   No high-priority defects remain unresolved.

4. **Analyze Defects**

   o   Record defects in a defect tracking tool (e.g., Jira).

   o   Categorize defects (Critical, High, Medium, Low).

5. **Generate Quality Reports**

   o   Document test results and defect trends.

   o   Use graphs (e.g., Pareto charts) to prioritize key quality issues.

6. **Implement Corrective Actions**

   o   Address defects identified during testing.

   o   Modify processes to prevent similar defects in future iterations.

https://thedigitalprojectmanager.com/projects/quality-management/quality-management-plan/

EX. NO: 7 ASSIGNING TASKS AND RESPONSIBILITIES USING PROJECT MANAGEMENT TOOLS

Aim:

To demonstrate the process of assigning tasks and responsibilities to team members using project management tools to ensure effective project execution and accountability.

- **Software/Tools**:
  - Trello (or an equivalent project management tool).
  - Internet connection to access the tool's web interface or app.
- **Team Roles**:
  - Project Manager: Creates and assigns tasks.
  - Team Members: Accept assigned tasks and update progress.

Part A: Setting Up the Project Management Tool

Steps

1. **Create a New Project**:
   - Log in to the tool.
   - Create a new project or board (e.g., "Website Development Project").
2. **Set Up Workflow Stages**:
   - Create lists (or columns) for the project workflow:
     - To Do
     - In Progress
     - Under Review
     - Completed
3. **Define Tasks**:
   - Break the project into smaller, actionable tasks.
   - Example Tasks:
     - Design homepage wireframe.
     - Develop homepage code.
     - Conduct usability testing.

Part B: Assigning Tasks and Responsibilities

Steps

1. **Add Team Members**:
   - Invite team members to the project/board.
   - Ensure each team member has access to the tool.
2. **Assign Tasks**:
   - Click on a task/card.
   - Add a responsible team member (or multiple members).

- o Example:
  - ▪ Task: "Develop homepage code" → Assigned to Alex.
3. **Define Deadlines**:
  - o Set due dates for tasks.
  - o Example:
    - ▪ Task: "Develop homepage code" → Due Date: January 5, 2024.

Add Descriptions and Subtasks:

- Provide a detailed task description.
- Add checklists for subtasks.
- Example:
  - o Task: "Develop homepage code."
    - ▪ Subtasks:
      - ▪ Write HTML structure.
      - ▪ Add CSS styling.
      - ▪ Implement JavaScript functionality.

Assign Priorities:

- Use labels or tags (e.g., High, Medium, Low priority).
- Example:
  - o Task: "Fix critical bugs" → High Priority.

Part C: Monitoring Task Progress
Steps

1. **Update Task Status**:
  - o Move tasks across workflow stages (e.g., from "To Do" to "In Progress").
  - o Example:
    - ▪ Task: "Design homepage wireframe" → Moved to "Completed."
2. **Track Progress**:
  - o Use reporting features or progress bars in the tool.
  - o Example:
    - ▪ 3 out of 5 tasks for the "Homepage Design" milestone are complete.
3. **Comment and Collaborate**:
  - o Use the comment section in task cards to communicate updates or issues.
  - o Example:
    - ▪ Comment: "Waiting for client feedback before finalizing."
4. **Review Deadlines**:
  - o Monitor overdue tasks and send reminders to team members.

Task	Assignee	Due Date	Status	Remarks
Design homepage	Priya	2024-01-03	Completed	Approved by

Task	Assignee	Due Date	Status	Remarks
wireframe				client.
Develop homepage code	Alex	2024-01-05	In Progress	HTML and CSS completed.
Usability testing	Rahul	2024-01-07	To Do	Scheduled for next week.

Result:

Using project management tools:
1. Tasks can be assigned with clarity and accountability.
2. Deadlines and priorities ensure timely execution.
3. Workflow visualization helps monitor progress effectively.

EX.NO: 8 USING COLLABORATION TOOLS FOR EFFECTIVE COMMUNICATION AND DOCUMENT SHARING.

Aim:

To demonstrate how to use collaboration tools for effective communication and document sharing,Using **Microsoft Teams** and **Google Workspace**, simulating a real-world scenario where team members collaborate on an e-commerce website development project.

Step 1: Set Up Microsoft Teams

Create a Team and Channels

1. **Create the Team**:
   - Open Microsoft Teams.
   - Click on **Teams** on the left sidebar.
   - Select **Join or Create a Team**.
   - Click **Create Team** and choose **Private** for a project-specific team.
   - Name the team **E-Commerce Website Development**.
2. **Create Channels**:
   - Under your team, create the following channels:
     - **General**: For general discussions and project updates.
     - **Design**: For design-related communication and mockup reviews.
     - **Development**: For discussions on coding, technical specs, and bugs.
     - **Testing**: For QA feedback, bug tracking, and test case creation.

Invite Team Members

1. **Invite Members**:
   - Click on the **More Options** (three dots) next to your team name.
   - Select **Add member**.
   - Add team members and assign them to relevant channels based on their roles (Design, Development, Testing).

Step 2: Set Up Google Workspace

Create a Shared Google Drive Folder

1. **Create a Shared Folder**:
   - Go to **Google Drive**.
   - Click on **New** and select **Folder**.
   - Name the folder **E-Commerce Website Docs**.
   - Right-click the folder and select **Share**.
   - Set the permissions to allow all team members to access, edit, or comment on documents.
2. **Create Subfolders**:

   Within the shared folder, create subfolders for different stages and documents:

- o **Designs**: For design mockups and assets.
- o **Development**: For code, technical documents, and API specifications.
- o **Testing**: For test plans, test results, and bug reports.
- o **Final Deliverables**: For completed project documents and reports.

Create Google Docs and Sheets

1. **Project Plan Document**:
    - o Create a **Google Doc** for the **Project Plan**.
    - o Outline key project milestones, deadlines, and team responsibilities.
    - o Share the link to the **Project Plan** in the **General** channel of Microsoft Teams.
2. **Project Timeline Spreadsheet**:
    - o Create a **Google Sheet** for the **Project Timeline**.
    - o Use columns to track tasks, deadlines, and team members assigned.
    - o Share the timeline in the **Development** channel on Teams.

Step 3: Use Teams for Communication

1. Post an Announcement in the General Channel

- **Project Manager's Post**:
    - o **Message**: "Welcome to the **E-Commerce Website Development** project. Please review the **Project Plan** in Google Docs. Our first sprint starts on Monday. Check out the **Project Timeline** for upcoming tasks!"
    - o **Action**: Share the **Google Doc** and **Google Sheet** links in the message for easy access.

2. Use Threads for Focused Conversations

- **Design Team Discussion**:
    - o In the **Design** channel, the designer shares the first version of the homepage mockup:
        - ▪ **Message**: "Here's the first draft of the homepage design. Please review and provide feedback. [Google Drive Link to Mockup]"
    - o Other members reply in the thread to discuss the design changes.
        - ▪ **Developer's Comment**: "Can we add a category dropdown in the navigation bar?"
        - ▪ **QA's Comment**: "Looks good! Make sure it's responsive for mobile."

3. Assign Tasks and Deadlines

- **Task Assignment**:
    - o **Project Manager**: "Alex, please start developing the homepage by reviewing the design in the **Design** channel. The deadline for the homepage is January 20th."
    - o **Message**: Post the task in the **Development** channel and link to the design mockup for easy reference.

Step 4: Use Google Workspace for Document Sharing and Collaboration

1. Share and collaborate on Google Docs

- **Google Docs - Project Plan**:
  - All team members can open and edit the **Project Plan** document in real time.
  - **Example**:
    - **Developer**: Adds a note: "Can we shift the checkout feature to Sprint 2?"
    - **Designer**: Replies, "Yes, that works. I'll update the design next week."

2. Track Tasks and Milestones with Google Sheets

- **Google Sheets - Project Timeline**:
  - The **Project Manager** updates the **Google Sheet** with new tasks.
  - Example:
    - Task: Develop Homepage
    - Assigned to: Alex
    - Due Date: January 20
    - Status: **In Progress**

Update Status:

- Team members update the status as tasks progress. If a task is delayed, it's marked in **red** with comments explaining the reason.

Step 5: Organize and Share Final Deliverables

1. Upload Final Documents to Google Drive

- **Final Design**:
  - Once the design is finalized, upload the **homepage mockup** to the **Designs** folder in Google Drive.
  - Share the link in the **Design** channel on Teams to notify the team.

2. Share the Final Report in Teams

- **Project Manager**:
  - Once the project is completed, share a final report document in the **Final Deliverables** folder.
  - Post in the **General** channel: "The project is complete. All final documents are uploaded to Google Drive. Please review the **Final Report**."

Step 6: Video Meetings for Syncing and Updates

- **Weekly Standups**:
  - Use **Microsoft Teams** to schedule a quick video standup meeting to review progress and discuss any blockers.
  - Example:
    - **Project Manager**: "Let's go over what we've accomplished this week and what needs attention."
    - **Designer**: "The homepage design is ready for review."
    - **Developer**: "I've finished coding the login page, but need feedback from QA."

Result:

In this demo, you've seen how **Microsoft Teams** and **Google Workspace** can be used together to:

- Facilitate communication through channels and real-time messaging.
- Share and collaborate on documents efficiently using Google Docs and Sheets.
- Track and manage project tasks and milestones through Google Sheets.
- Organize and store project files in Google Drive for easy access.

EX.NO:9 IMPLEMENTING AGILE METHODOLOGIES FOR PROJECT MANAGEMENT

Steps:

1. **Set the project vision and scope**

2. **Build the product roadmap**

3. **Create a release plan**

4. **Plan sprints**

5. **Use daily standups to keep the team on track**

6. **Conduct sprint reviews**

Here's how you can implement Agile in a lab setting:

Project Scenario

Develop a simple application, such as a "Task Management System," using Agile practices.

Step-by-Step Lab Activities

1. **Define Project Objectives**:

   o Create a backlog of tasks (e.g., user login, task creation, task deletion).

2. **Assign Roles**:

   o Form small teams and assign roles like Product Owner, Scrum Master, and Developers.

3. **Plan Sprints**:

   o Divide tasks into 2-week sprints.

4. **Daily Stand-ups**:

   o Conduct 10-minute meetings to discuss progress.

5. **Use Agile Tools**:

   o Visualize tasks on Trello or a physical Kanban board.

6. **Conduct Reviews**:

   o At the end of each sprint, demonstrate the completed tasks to stakeholders.

7. **Retrospectives**:

   o Discuss successes and areas of improvement.

Assessment Metrics

- Velocity: Number of story points completed in a sprint.

- Burn Rate: Effort spent versus planned effort.

- Defect Rate: Quality of delivered increments.

Tools and Resources

- **Software**:

  o Jira, Trello, Monday.com for project tracking.

  o GitHub for version control.

Result:

Implementing Agile methodologies enhances project delivery by promoting collaboration, flexibility, and continuous improvement.