Project Planning Template

Project Planning Logic

A Sprint fixed period or duration in which a team works to complete a set of tasks

An Epic is a big task or project that is too large to complete in one sprint. It is broken down into smaller tasks (stories) that can be completed over multiple sprints.

A Story is a small task . It is part of an Epic.

A Story Point is a number that represents how much effort a story takes to complete. (usually in form of Fibonacci series)

1. Very Easy task
2. Easy task
3. Moderate task
4. Difficult task

Sprint 1: (5 Days)

Data Collection

- Collection of Data 2
- Loading Data 1

Data Preprocessing

- Handling Missing Values 3
- Handling Categorical values 2

Sprint 2 (5 Days)

Model Building

- Model Building 5
- Testing Model 3

Deployment

- Working HTML Pages 3
- Flask deployment 5

Project Planning Template

1. Define Clear Objectives:

    - Why: To ensure everyone understands what the project will achieve.
    - Logic: Without clear goals, the project scope will drift. For example, your goal is to digitize garage operations and improve customer satisfaction.

2. Establish Scope:

    - Why: To set boundaries on what's included and excluded.
    - Logic: Scope clarity prevents scope creep — extra features or tasks that can derail deadlines and budgets. For example, "Online booking" is in scope; "Physical garage renovations" is out of scope.

3. Break Down Deliverables:

    - Why: To make big goals manageable.
    - Logic: Deliverables split the work into chunks — like requirements, design, development, testing, and deployment — so progress is measurable and responsibilities are clear.

4. Create a Milestone Timeline:

    - Why: To set realistic start and end dates.
    - Logic: Divides the project into phases with deadlines. For your short timeline (25-27 June 2025), tasks must overlap efficiently — e.g., requirements and design on Day 1, development and testing on Day 2, deployment on Day 3.

Milestone	Start Date	End Date	Owner
Requirements Gathering	25 Jun 2025	25 Jun 2025	Devi sri priya
Design Phase	25 Jun 2025	25 Jun 2025	Devi sri priya
Development Phase	25 Jun 2025	26 Jun 2025	Devi sri priya
Testing & QA	26 Jun 2025	26 Jun 2025	Devi sri priya
Deployment & Go Live	27 Jun 2025	27 Jun 2025	Devi sri priya
Project Closure & Handover	27 Jun 2025	27 Jun 2025	Devi sri priya

5. Assign Roles & Responsibilities:

- Why: So everyone knows who does what.
- Logic: Clear ownership prevents duplication of work and missed tasks. For example, the developer codes while the QA tests in parallel.

Role	Responsibility	Person
Project Manager	Planning, coordination, reporting	Devi sri priya
Developer(s)	Coding, unit testing	Devi sri priya
Designer(s)	UI/UX mockups, wireframes	Devi sri priya
QA Tester(s)	Test planning, execution, bug tracking	Devi sri priya
Client Stakeholder	Approvals, feedback, requirements	Devi sri priya

6. Plan for Risks:

- Why: To prepare for what could go wrong.
- Logic: You identify high-impact risks (like scope creep or resource unavailability) and mitigation strategies (like clear sign-offs or backup resources).

Risk	Impact	Mitigation Plan
Scope creep	High	Define scope clearly & get sign-offs
Resource availability	Medium	Backup resources identified
Integration delays	Medium	Early vendor coordination
Data loss/security breaches	High	Backup plan, strong security protocols