

How to become project manager, product owner or team leader

⚠ Benefits of being a project manager, product owner or team leader

- shape projects, products & organizational outcomes
- mentorship & skill development
- broader perspective on software development
- create a supportive work culture
- enhances your professional reputation within the organization
- greater influence on the team success
- expand your skill set beyond technical expertise
- transferable skills
- stepping stones towards director or head of department position

⚠ project manager's responsibilities

- setting project goals & objectives
- planning & resource allocation
- risk management
- communication and collaboration

⚠ product owner's responsibilities

- defining the product vision
- prioritizing features & managing product backlog
- stakeholder management
- acceptance & validation

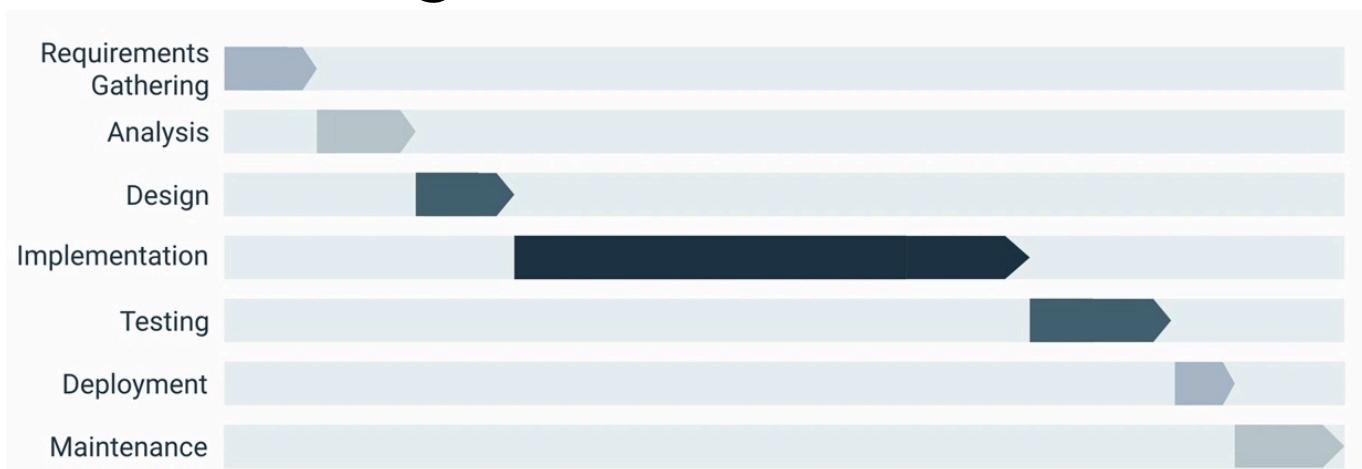
⚠ team leader's responsibilities

- team development and motivation
- task assignment and coordination
- facilitating communication & removing obstacles
- performance evaluation & feedback

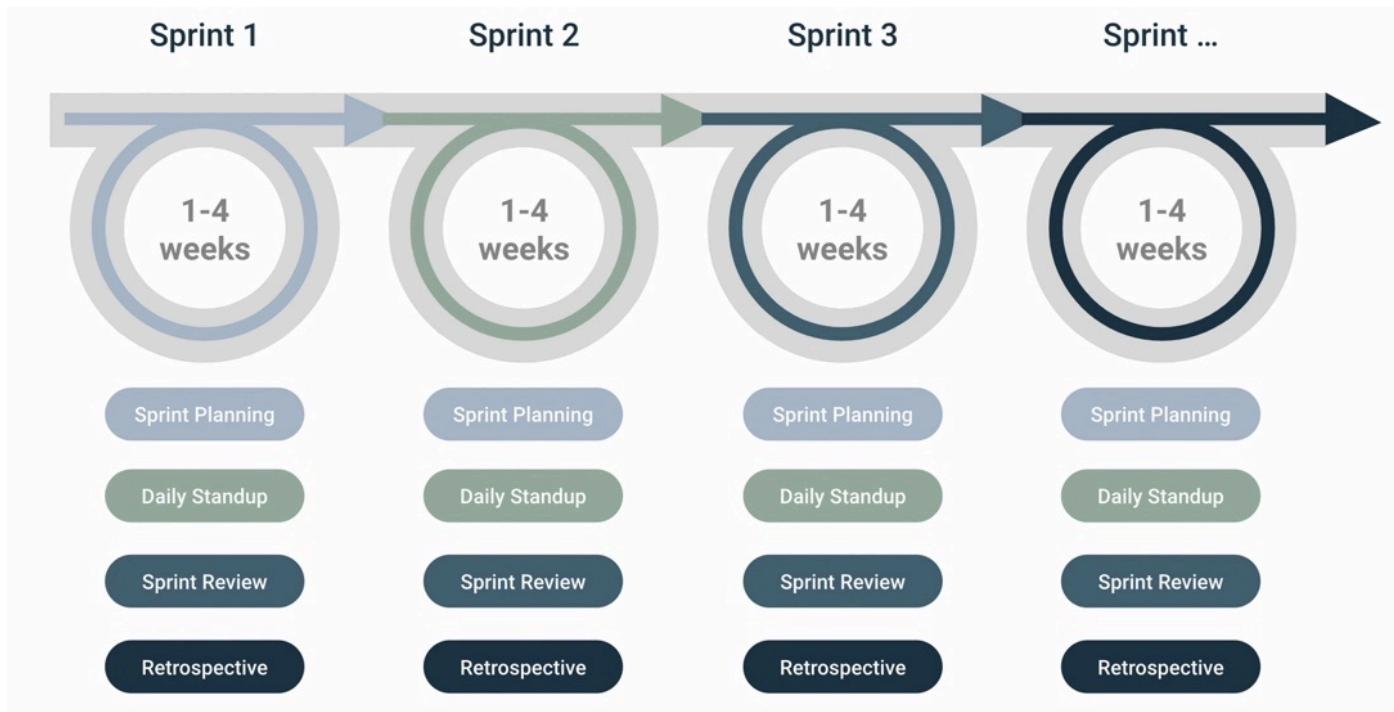
⚠ involvement in technology & business

	Technology	Business
Project Manager	Low	High
Product Owner	Medium	Medium
Team Leader	High	Low

⚠ Waterfall: emphasizing structure & predictability



⚠ Agile: Embracing flexibility and adaptability



⚠ benefits of the project charter

- clarity & alignment
- scope management
- resource allocation
- risk mitigation

⚠ importance of having a project charter

- Project Focus
- efficient resource allocation
- team collaboration
- communication tool

⚠ project charter should include

- ① project Name
- ② project Description
- ③ project objectives
- ④ scope
- ⑤ key stakeholders
- ⑥ timeline
- ⑦ budget
- ⑧ risks
- ⑨ assumptions
- ⑩ success criteria
- ⑪ project approval

⚠ importance of requirements gathering & analysis

- understanding stakeholder needs
- defining project scope
- setting clear objectives
- risk identification and mitigation

⚠ requirements gathering techniques

- interviews
- surveys
- workshops & focus groups
- observation
- prototyping & mock-ups
- document analysis

⚠ benefits of the work breakdown structure

- clarity and scope definition
- task identification and organization
- resource allocation
- project tracking and monitoring

⚠ process of crafting a robust work breakdown structure

- define project scope
- identify major phases
- decompose tasks
- engage stakeholders
- validate & review
- hierarchical structure
- code of accounts
- maintain flexibility

⚠ benefits of resource estimation

- aligning expectations
- risk mitigation
- decision-making
- stakeholder confidence

⚠ techniques involved in estimating project time

- task dependencies
- historical data
- expert judgement
- parallel & sequential tasks
- buffer time
- iterative refinement

⚠ the significance of project monitoring

- performance tracking
- risk management
- quality assurance
- adaptability & flexibility
- reporting & communication
- continuous improvement

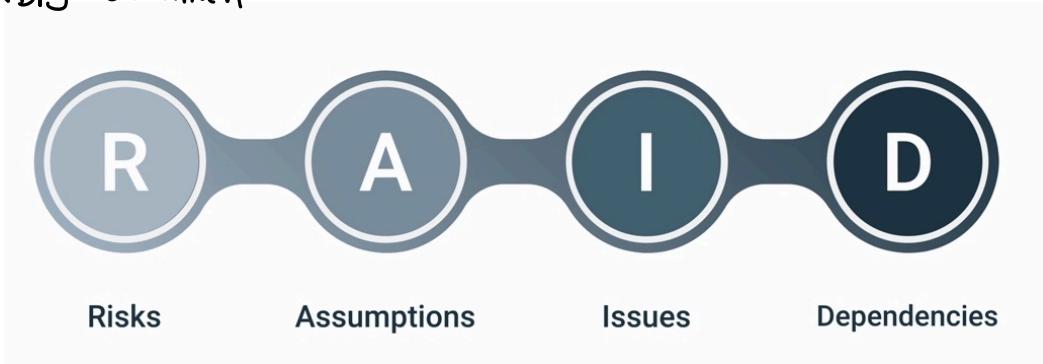
⚠ techniques for effective scope management

- formal change control process
- prioritize change requests
- regular communication & reporting
- continuous monitoring & control
- scope verification & validation

⚠ strategies for identifying and assessing risks

- comprehensive risk identification
- risk assessment & prioritization
- stakeholder involvement
- risk registers & documentation
- risk categorization
- risk workshops & analysis

⚠ The RAID document



⚠ benefits of using the RAID document

- enhanced communication
- proactive risk management
- issue resolution
- efficient collaboration
- improved decision-making
- project documentation & auditing

△ Quality assurance encompasses a set of processes & activities aimed at ensuring that the software being developed meets specified requirements to enhance the overall quality of the software

△ Testing involves systematically verifying & validating the software to identify & fix any issues before deployment

△ Importance of QA & testing

- minimizing risk
- customer satisfaction
- cost efficiency
- compliance & standards

△ QA metrics & reporting

- defect density
- test coverage
- defect closure rate
- test execution progress

△ Project closure details

- ① evaluation & learning
- ② lessons learned
- ③ resource optimization
- ④ stakeholder satisfaction
- ⑤ archives

⚠ Agile Values

- individuals and interactions over processes & tools
- working software over comprehensive documentation
- customer collaboration over contract negotiation
- responding to change over following a plan

⚠ Agile principles

1	Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.	7	Working software is the primary measure of progress.
2	Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.	8	Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
3	Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.	9	Continuous attention to technical excellence and good design enhances agility.
4	Business people and developers must work together daily throughout the project.	10	Simplicity--the art of maximizing the amount of work not done--is essential.
5	Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.	11	The best architectures, requirements, and designs emerge from self-organizing teams.
6	The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.	12	At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

⚠ Agile frameworks

- ① Scrum
- ② Leanken
- ③ Lean
- ④ XP
- ⑤ Nexus
- ⑥ Less
- ⑦ SAFe

⚠ Scrum pillars

- transparency
- inspection
- adaptation

⚠ Scrum values

- focus
- courage
- commitment
- openness
- respect

⚠ Scrum roles

- product owner
- scrum master
- development team

⚠ product owner responsibilities

- defining the product vision
- managing product backlog
- prioritization & decision-making
- collaborating with stakeholders
- sprint review & acceptance

A How to become a great product owner

- strong communication skills
- decision-making & prioritization
- empathy & user focus
- business awareness
- collaborative spirit
- resilience & adaptability

A challenges & tips for success for POs

- balancing priorities
- incomplete requirements
- avoiding micro management
- handling stakeholder expectations
- managing scope creep

A key components of sprint planning

- product backlog review
- sprint goal
- capacity planning
- definition of done (DoD)

A Key elements of the daily scrum

- ↳ time-boxed
- ↳ focus

A Key components of the sprint review

- product increment presentation
- sprint goal review
- feedback & discussion
- adaptation & planning
- definition of done validation

A Key elements of the sprint retrospective

- ↳ time for introspection
- meeting topics
- inspect and adapt

A User Story template

↳ As a "user",
I want "an action",
so that "a benefit"

⚠ when to choose which approach

Story points	Time-based Estimations
requirements are likely to change or be uncertain	there's a need for precise deadline planning
emphasizing value delivery is crucial	historical data & consistent velocity are available
collaborating & shared understandings are important	resource allocation is a primary concern
tasks vary in complexity	tasks have a predictable & repeatable nature

⚠ Velocity

↳ team's capacity for delivering work within a sprint

⚠ estimation techniques

- planning poker
- T-shirt sizing
- bucket system
- affinity estimation

⚠ prioritization techniques

- MoSCoW method
- weighted shortest job first
- lcano model
- value - effort matrix
- cost of delay
- opportunity scoring

⚠ S.M.A.R.T goals

- specific
- measurable
- achievable
- relevant
- time-bound