

Product Owner	
	VISION
TASKS	<ul style="list-style-type: none"> Communicate with Stakeholders Understand business needs Create product roadmap Define business KPI's Align product accordingly

PRIMARY LIAISON	
TASKS	<ul style="list-style-type: none"> Gain buy-in from stakeholders Clear instructions to SM

ANTICIPATE	
TASKS	<ul style="list-style-type: none"> Analyze markets Anticipate client needs Customer journey maps

PRIORITIZE	
TASKS	<ul style="list-style-type: none"> Consider needs and objectives Juggle, scope, budget and time Map dependencies Manage backlog Prioritize items

MONITOR	
TASKS	<ul style="list-style-type: none"> Contribute to planning Contribute to execution Contribute to review Improve process Inspect iterations Determine acceptance

Scrum Master	
	ENCOURAGE
TASKS	<ul style="list-style-type: none"> Face to face communication Self organization Accountability Openness in standups Transparency in ceremonies Adaptability in change Autonomy in troubleshooting

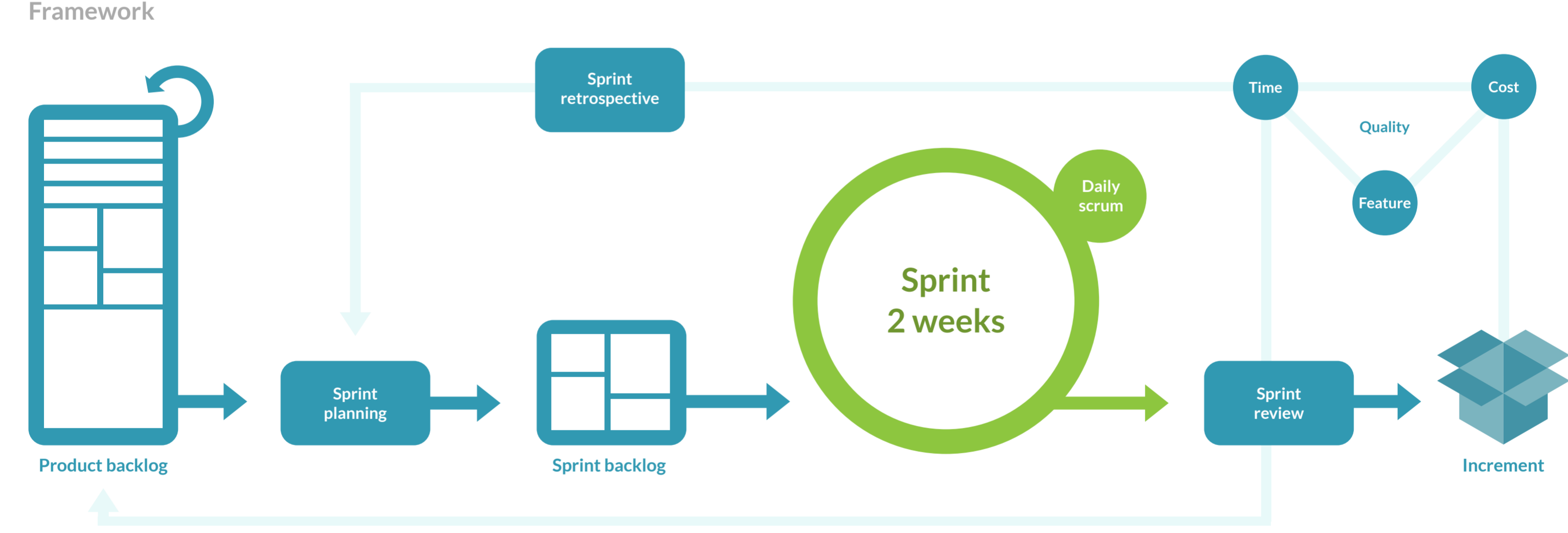
FACILITATE	
TASKS	<ul style="list-style-type: none"> All meetings and ceremonies Collaboration with stakeholders Team release planning Suitable definition of done Maintaining sustainable pace Team bonding activities

LEARN AND SHARE	
TASKS	<ul style="list-style-type: none"> Self-learn on agile Coach and consult on agile Interact with other SM's Regular feedback to team Help team monitor metrics

REWARD AND PROTECT	
TASKS	<ul style="list-style-type: none"> Protect team from obstacles Mediate on conflicts Get rid of impediments Appriciate good work Show pride in deliverables Celebrate success Make moments memorable

REFLECT	
TASKS	<ul style="list-style-type: none"> Visualize metrics Report to management Promote retrospective Maintain collaboration tools Promote retrospective Give outside observations

Ceremonies					
	1	2	3	4	
	Backlog Refinement	Sprint planning	Daily standup	Sprint review	Sprint retrospective
	2 Hours	2 Hours (2 weeks)	15 min	2 Hours (2 weeks)	90 min (2 weeks)
TASKS	<ul style="list-style-type: none"> Remove irrelevant user stories Presents upcoming ideas Define definition of done Discuss minimal viable product Break-down if possible High level estimates Prioritize items by ROI 	<ul style="list-style-type: none"> Discuss goal for the sprint Discusses any sprint deviations Present the sprint velocity Elicit and clarify team capacity Assesses areas of likely risk Review the definition of done Present proposed sprint backlog Assess effort and dependencies Call for group consensus 	<ul style="list-style-type: none"> What have you been working on ? What are you going to work on ? Are there any dependencies ? 	<ul style="list-style-type: none"> Recap the goals of the sprint Explain business value of items Demo each feature Explain goals vs. progress Explain goals not meet Will items overlap next sprint Take question & give answers 	<ul style="list-style-type: none"> What did we do right What did we do wrong What should we start doing What should we stop doing What can we do to improve
RACI	Team - (Responsible) ScrumMaster - (Accountable) Product Owner - (Responsible) Stakeholders - (Informed)	Team - (Responsible) ScrumMaster - (Accountable) Stakeholders - (Informed)	Team - (Responsible) ScrumMaster - (Accountable) Stakeholders - (Informed)	Team - (Responsible) ScrumMaster - (Accountable) Product Owner - (Responsible) Stakeholders - (Informed)	Team - (Responsible) ScrumMaster - (Accountable) Product Owner - (Informed)



PLANNING AND LEADERSHIP	CULTURE	DELIVERY	TECHNICAL HEALTH
<ul style="list-style-type: none"> Vision clarity Backlog management Prioritization Stakeholder engagement Risk management Leadership confidence Sponsor engagement Tracking and reporting 	<ul style="list-style-type: none"> Creativity and innovation Sustainable pace Trust and respect Empowerment Cross-functional collaboration 	<ul style="list-style-type: none"> Escaped Defects Failed Deployments Net Promoter Score (NPS) Lead Time Cycle Time (Control Chart) Sprint, Epic and release burndown Cumulative flow Velocity 	<ul style="list-style-type: none"> Build infrastructure Automated testing Deployment Documentation Technical debt Technical roadmap Experiemental infrastructure

PRINCIPLES	TECHNIQUES	METHODS	CONCEPTS
<ul style="list-style-type: none"> Focus on the business need Deliver on time Collaborate Never compromise quality Build incrementally from firm foundations Develop iteratively Communicate continuously and clearly Demonstrate control 	<ul style="list-style-type: none"> User stories and backlogs Moscow prioritization Goal and functional decomposition Personas Prototypes Planning poker & estimation techniques Velocity and burndown charts 	<ul style="list-style-type: none"> Agile Scrum Methodology Lean Software Development Kanban Extreme Programming (XP) Crystal Dynamic Systems Development Method Feature Driven Development (FDD) 	<ul style="list-style-type: none"> Transparency Inspection Adaption