

Agile Project Management

Prepared by Eng. Ahmed Hussein,
MBA, PfMP®, PMP®, CMP APMG

PMG 622
Master of Project Management

Lecture 4



**Strengths and Weaknesses
of the Agile Approach**

Lecture 4

PMG 622

Agile Project Management

Prepared by
**Eng. Ahmed Hussein, MBA, PfMP®,
PMP®, CMP APMG**

Lecture
04

Learning Objectives

Learn Agile techniques, strengths and weaknesses for:

1. Scope Management
2. Time Management
3. Cost Management

Key Points from Previous Lecture

Traditional and Agile PM Life Cycles have many similarities and also differences

Selection of the RIGHT Life Cycle depends on Degree of Change, Frequency of Delivery, and Context of Project

Traditional Life Cycle can be Incremental (ex. Rolling out a system to many branches)

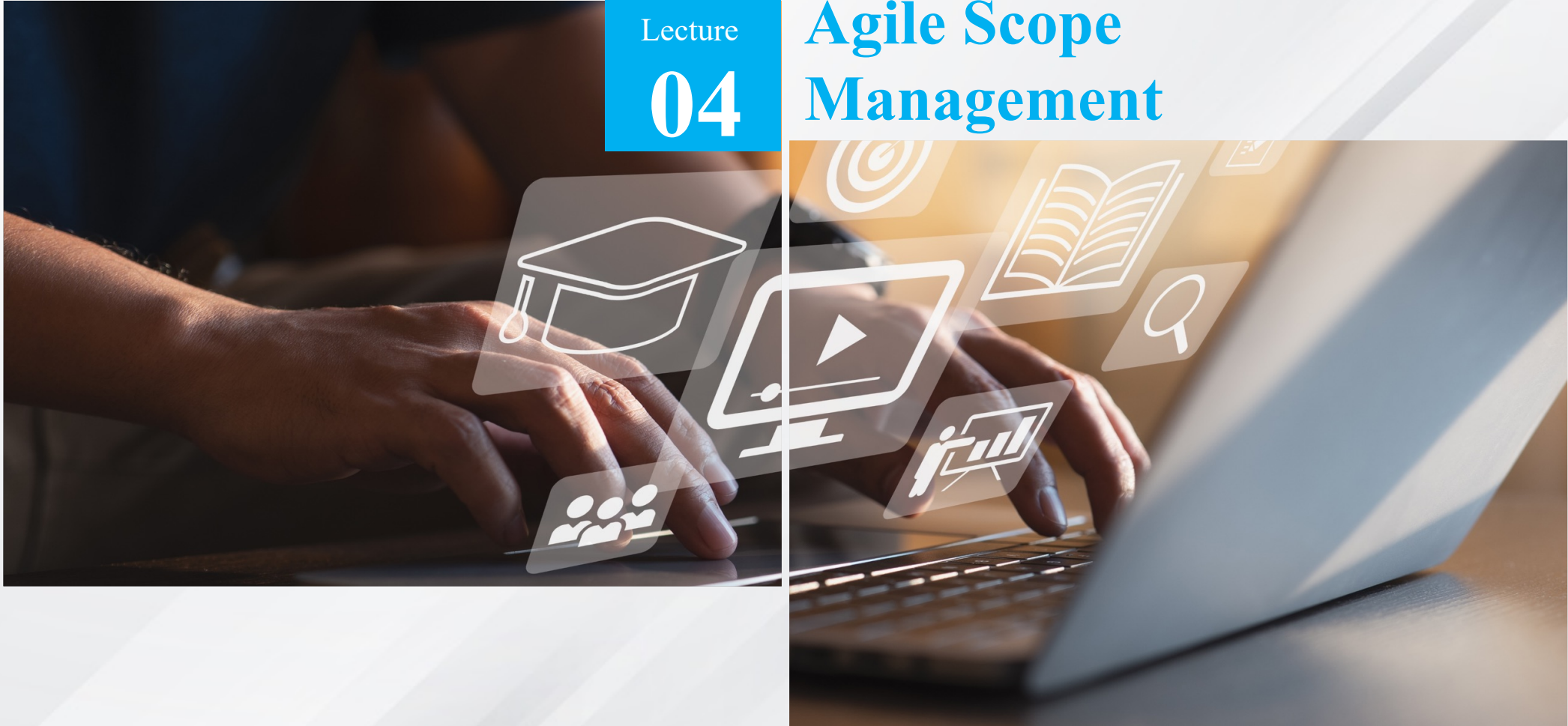
Agile Life Cycles are usually Incremental, Iterative, and Adaptive

Each Life Cycle has its Strengths, weaknesses, benefits and drawbacks

Lecture

04

Agile Scope Management



Lecture

04

Agile Projects have Variable Scope

- Traditional project management treats changing requirements as a sign of failure in upfront planning
- Variable Scope through welcoming change helps to immediately and incrementally:



Incorporate Learning and
Feedback



Create better Products

Top Agile Scope Management Characteristics (1/2)

#	Agile Scope Management
1	Only High-Level requirements are gathered at the beginning of the project
2	Requirements are gathered and refined throughout the project as knowledge about customer needs and project realities increase
3	Change is viewed as a positive way to improve the product and Late changes in the project are the most valuable ones
4	The Product Owner is the responsible person for the project scope (not the Scrum Master), determines the value and priority of new requirements and adds those requirements to the product backlog.
5	There is no need for Change Management processes, as it is already an inherent part of Agile processes

Top Agile Scope Management Characteristics (2/2)

#	Agile Scope Management
6	New features with high priority don't necessarily cause budget or schedule slip; they simply push out the lowest-priority features
7	The most valuable features first to guarantee their inclusion and to ship those features as soon as possible
8	Less valuable features might never be created, which may be acceptable to the business and the customer after they have the highest-value features
9	Scope is assessed and there is an opportunity to include new requirements or modify/remove existing with every Iteration/Sprint

Who can suggest a new product feature on an agile project?



Feedback from User Community

Groups or people who are given an opportunity to preview the product



Managers and Business Stakeholders

Can see new market opportunities or threats and gave insight into long-term organizational strategies and changes



Scrum Master and/or Team

They are learning more about the product every day, and are closest to the working product



Product Owner

who often knows the most about the product and the stakeholders' needs



Product Owner

determines the value and priority of new requirements and adds those requirements to the product backlog.

Managing Scope Changes

Assess

Assess if the new requirement should be part of the product, release, or sprint



Product Owner

Estimate

Estimate the Effort for the new requirement



**Product Owner, Scrum Master
and/or Team**

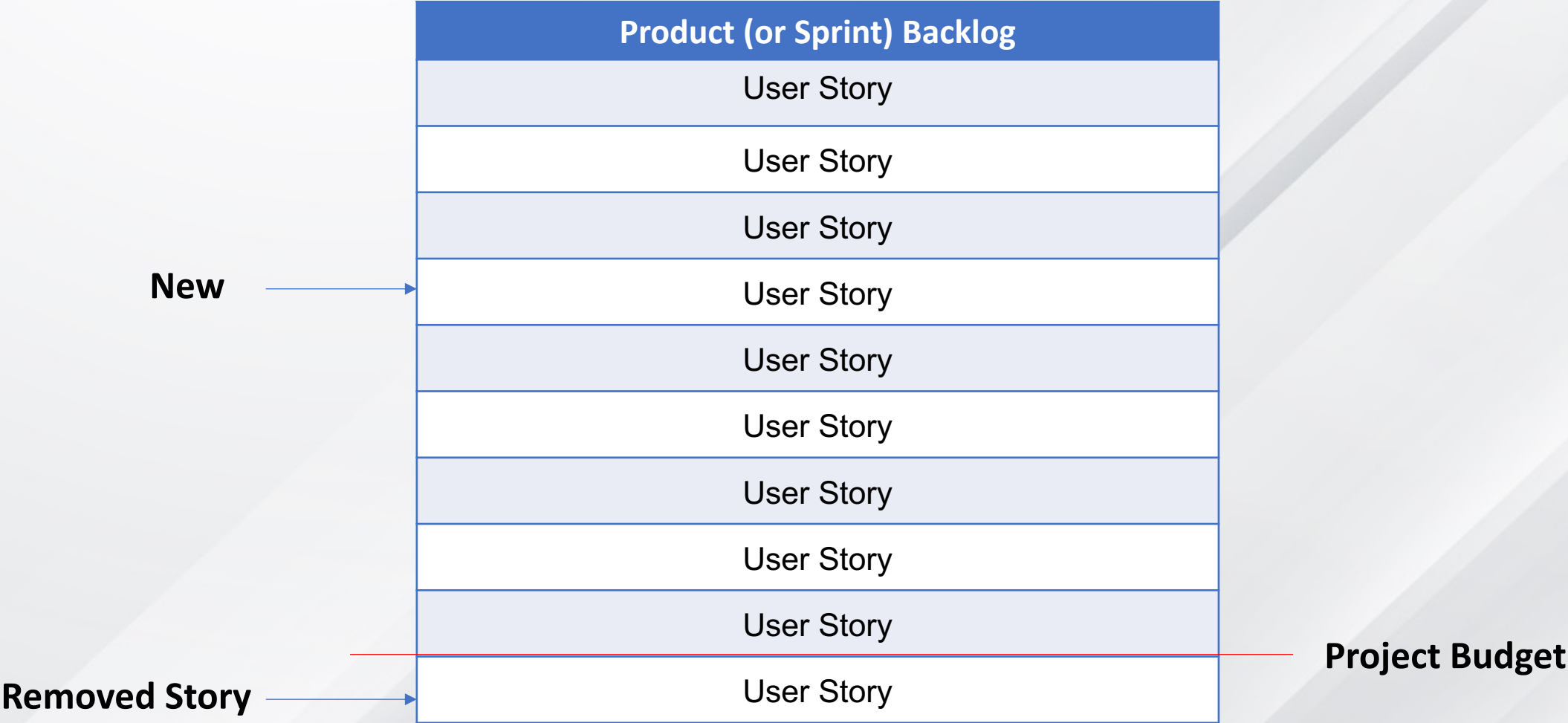
Prioritize

Prioritize against other requirements in the product backlog and add to the product backlog, in order of priority.



Product Owner

Adding a New Requirement to the Backlog



Lecture

04

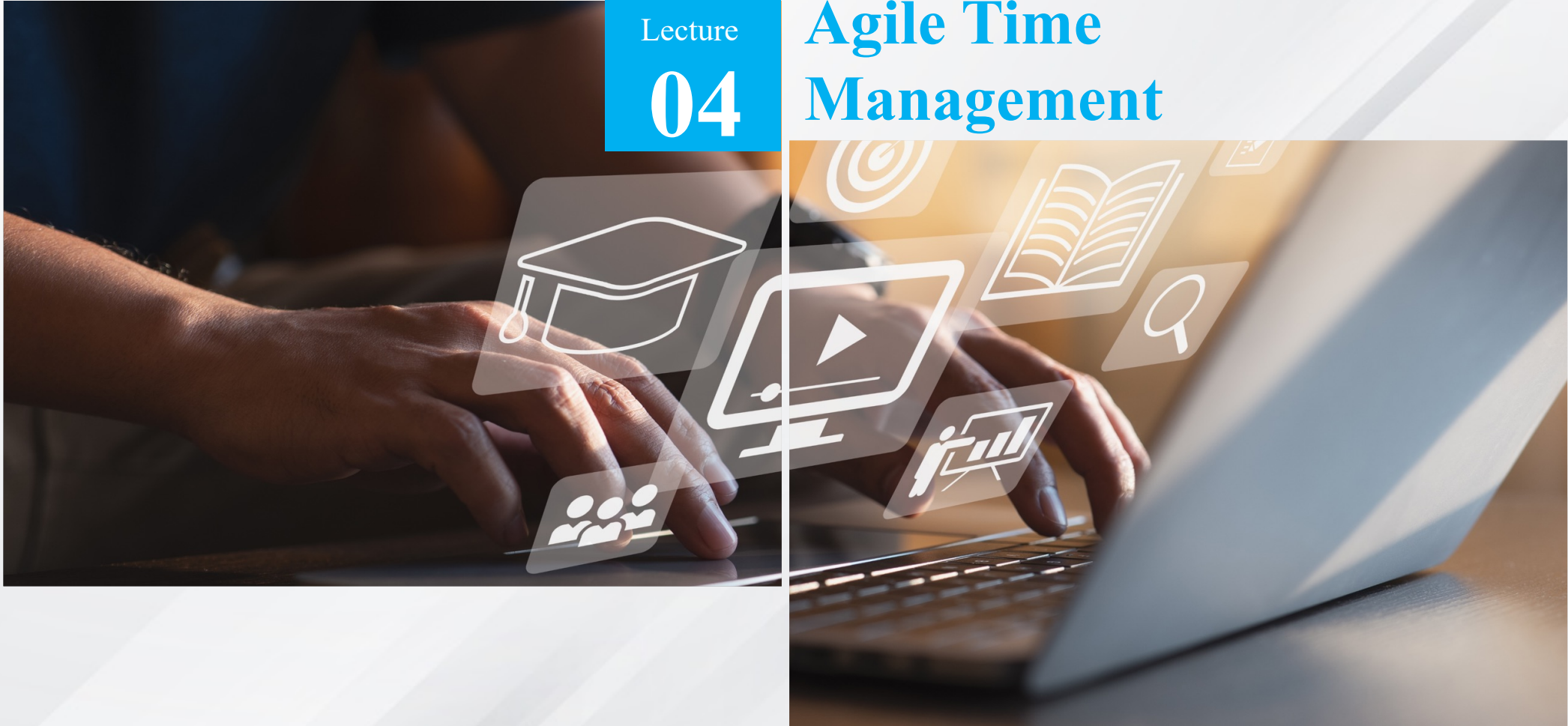
Documents used in Agile Scope Management

Document	Used for
Vision Statement	High level product requirements and used to judge whether new features belong in the scope for the current project.
Product Roadmap	- Product scope is part of the product roadmap. - Requirements are at a feature level - Good for business conversations about what it means to realize the product vision - Provides visual communication of the features included in the project
Product Backlog	- Detailed requirements in form of features or User Stories - Contains all scope features, and new added features - New, high-priority features push lower-priority features down on the product backlog
Sprint Backlog	- Contains the user stories that are in scope for the current sprint - After the sprint-planning meeting, only the team can modify the sprint backlog.

Lecture

04

Agile Time Management



Agile Projects usually have Fixed Deadlines

Traditional Project
Management

Determine Requirements



Determine Schedule based
on how much time to
achieve those Requirement

Agile Project
Management

Fixed Deadline



Create as much features as
you can before the deadline

Factors Impacting Agile Project Duration

Assigned Deadline	Budget Considerations	Features Completed
		
Agile projects may have a specified fixed date for business reasons. Ex. Getting a product to market before a shopping season Project works to create as much functionality as possible from project start until the end date.	Agile projects may have budget considerations that affect the amount of time a project will last. Ex., a \$1,600,000 budget, for a project that costs \$20,000 a week, the project will be able to last 80 weeks to create and release as much functionality as possible.	Agile projects may last only until enough functionality is complete. Project teams may run sprints until the requirements with the highest value are complete, and then determine that the lower-value requirements aren't necessary.

Top Agile Time Management Characteristics

#	Agile Time Management
1	Planning is done on strategic (product roadmap, release plan, product backlog) and tactical (iteration/sprint planning, daily meetings) levels
2	Team assess and reassess how much work they can complete in a given time frame (using velocity)
3	Team works in iterations/sprints and complete all the work on the highest-priority, highest-value requirements first, thus reducing overall project risk
4	Iterations/sprints are timeboxed (fixed duration), unfinished work is returned to product backlog for rescheduling
5	Execution starts early (as soon as iteration/sprint 1) with minimal planning and design time, usually resulting in shorter project time compared to traditional approach

Lecture

04

Agile Planning



How big is the effort?

How quickly can we get it done?

When can we do it?



The Agile Team



The Agile Team



The Agile Team

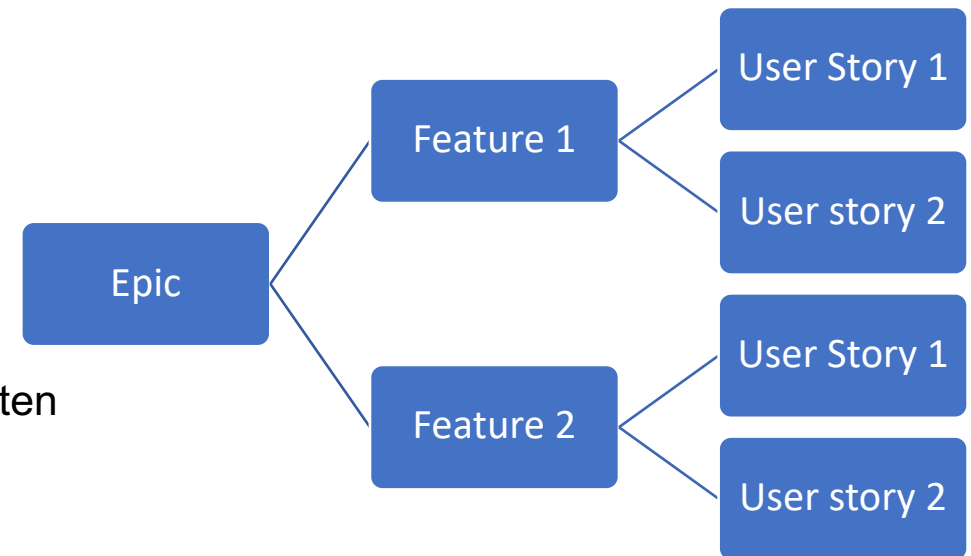
The Agile team is responsible for all steps of Planning (not the Product Owner nor the Scrum Master)

Lecture

04

Decomposing Requirements

- **Epics:** the highest level of work, represent large, complex goals that will take multiple sprints to complete. Epics are typically broken down into smaller, more manageable features.
- **Features:** smaller pieces of work that can be completed in a single sprint. Typically, specific and measurable, and they should deliver value to the customer.
- **User stories:** the smallest unit of work in agile. They are written from the perspective of the user, and they describe a specific task or function that the user wants to be able to perform.



Estimating using Story Points

Story Points:

An abstract unit of measure that represents the effort, complexity, and risk required to complete a user story. Story points are relative

- Typically use a simple scale to estimate story points, such as 1, 2, 3, 5, 8, 13, 20, 40, 100.
- Story points is a measure of EFFORT NOT TIME
- Story points size is determined by the team
- Story points are all inclusive (include time for documentation and testing)
- Used to:
 - Estimate the effort required to complete a sprint.
 - Prioritize stories in the product backlog.
 - Track their progress during a sprint.
 - Identify bottlenecks and improve their velocity over time

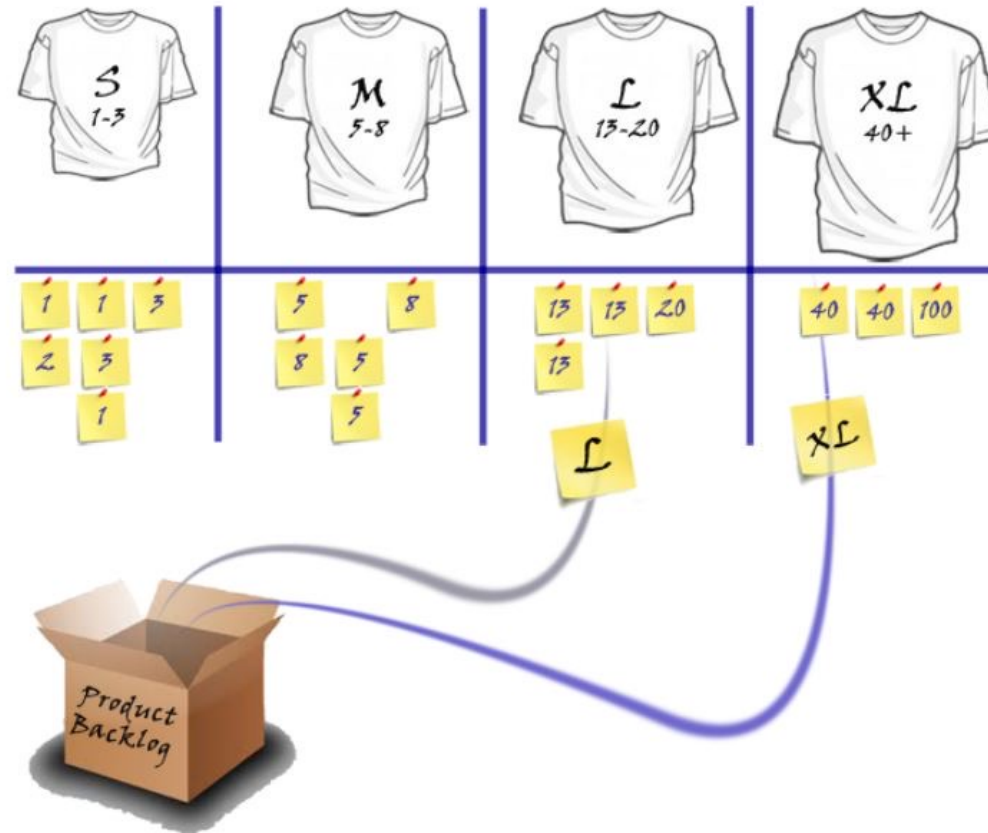
Lecture

04

Estimating using T-Shirt Sizing

of Story Points

User Stories



Calculating Velocity

- At the beginning of the project, the team estimates initial velocity
- Velocity is then calculated based on actual performance
- Velocity is measured in story points

$$\text{Current Velocity} = \frac{\text{Actual Velocity Iteration 1} + \text{Actual Velocity Iteration 2} + \dots + \text{Actual Velocity Iteration N}}{\text{Total Number of Iterations}}$$

$$\text{Current Velocity} = \frac{10+12+13}{3} = 11.6$$

Iteration/ Sprint #	Actual Velocity (in story points)
1	10
2	12
3	13
Current Velocity	11.6

Lecture
04

Iteration/Sprint Planning using Velocity

- Example:
- After 4 Iterations/Sprints, Current Velocity = 15

Velocity
Threshold

#	Description	Estimate (in Story Points)
1	User Story 10	3
2	User Story 11	7
3	User Story 12	5
4	User Story 13	4
5	User Story 14	6

Product Backlog after
Iteration/Sprint 4

#	Description	Estimate (in Story Points)
1	User Story 10	3
2	User Story 11	7
3	User Story 12	5
	Total Story Points	15

Iteration/Sprint 5 Backlog

Lecture 04

Documents used in Agile Time Management

Document	Used for
Product Roadmap	• A prioritized visual view of the high-level requirements and schedule • Usually does not have specific dates, it will have general date ranges for groups of functionality and will allow an initial timeline for bringing the product to market
Product Backlog	• A complete list of all currently known product requirements • The requirements in product backlog will have estimated story points. • After knowing team's velocity, a realistic project end date can be determined using the total number of story points in the product backlog
Release Plan	• Contains a release schedule for requirements • Has a target release date for a specific goal (epic or set of requirements) • Team plan and work on only one release at a time.
Sprint Backlog	• Contains the user stories that are in scope for the current sprint • During your sprint-planning meeting, you convert estimated story points to time. • In Sprint Retrospective, calculate velocity for that sprint based on finished story points

Lecture

04

Agile Cost Management



Top Agile Cost Management Characteristics

#	Agile Cost Management
1	Project schedule, not scope, has the biggest effect on cost
2	Projects may have a fixed cost and a fixed amount of time, and then complete requirements that fit into those costs and timeframe
3	Product owners often secure project funding after the product roadmap stage is complete, not before project start as in traditional approach
4	Project teams can replace lower-priority requirements with new, equivalently sized high-priority requirements with no effect on time or cost
5	Project teams can release working, revenue-generating functionality early that can spend on future project activities, creating a self-funding project.

Agile Project Costs

```
graph TD; A[Agile Project Costs] --> B[Fixed Costs]; A --> C[Per Sprint Costs];
```

Fixed Costs

- Fixed Costs (doesn't change with changing the # of Sprints in a project)
- Usually, One-time costs
- Can be paid upfront or along the project
- Examples:
 - Hardware Costs
 - Software Licenses
 - Training Costs
 - Hosting Costs

Per Sprint Costs

- Variable costs (depends on # of Sprints in a project)
- Examples:
 - Salaries
 - Rent
 - Utilities
- Calculated per Sprint (for ex. 2 weeks), i.e. monthly salaries divided by 2 sprints/month

Lecture

04

Creating an Initial Cost Estimate

- Estimate total # of Sprints from Product Roadmap, Release Plan
- Determine:
 - Fixed Costs
 - Per Sprint Costs

$$\text{Initial Cost Estimate} = \text{Fixed Costs} + (\text{Per Sprint Cost} * \text{\# of Sprints})$$

Managing a Fixed Budget

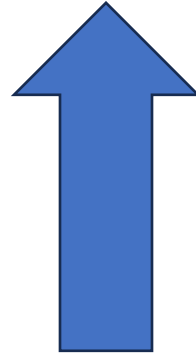
- Fixed Budget is when you are given a fixed amount for the project
- Determine Per Sprint Cost
- Determine Fixed Cost
- Determine how many Sprints you can do in the project:

$$\text{Maximum \# of Sprints} = \frac{(\text{Fixed Budget} - \text{Fixed Costs})}{\text{Per Sprint Cost}}$$

Lecture

04

How to Reduce Project Costs



Increase Velocity, as increased productivity will reduce # of sprints needed



Eliminate low priority features will reduce # of sprints needed

Lecture

04

Example on Managing Fixed Budget

- A project with a fixed budget of \$ 200,000
- The Project has:
 - Fixed Cost = \$ 0
 - Per Sprint Costs = \$20,000

$$\# \text{ of Sprints} = \frac{(\$ 200,000 - 0)}{\$ 20,000} = 10 \text{ Sprints}$$

Example on Managing Fixed Budget

- The Project is estimated to have 200 story points

$$\text{Velocity} = \frac{200}{10} = 20 \text{ Story points/Sprint}$$

- If new requirements are added making the project estimate to be 300 story points
- What is the Velocity you need to reach to finish the project on budget

$$\text{Velocity} = \frac{300}{10} = 30 \text{ Story points/Sprint}$$

Lecture 04

Self-Funding Projects

- A project that generates sufficient revenue to spend on further project activities
- Self –Funding projects enables continuous adding of features that deliver value

Self-Funding
Point

Sprint #	Cost	Cumulative Project Cost	Deliverable
Fixed Cost	\$ 40,000	\$ 40,000	
1	\$10,000	\$ 50,000	Release 1
2	\$ 10,000	\$ 60,000	Release 2
3	\$ 10,000	\$ 70,000	Release 3
4	\$ 10,000	\$ 80,000	Release 4
5	\$ 10,000	\$ 90,000	Release 5

Expenses

Release #	Income	Cumulative Income
-	-	-
Release 1	\$ 20,000	\$ 20,000
Release 2	\$ 20,000	\$ 40,000
Release 3	\$ 20,000	\$ 60,000
Release 4	\$ 20,000	\$ 80,000
Release 5	\$ 20,000	\$ 100,000

Income

Lecture

04

Documents used in Agile Cost Management

Document	Used for
Product Vision	Will usually have the project budget when it is fixed for the project
Product Roadmap	- A prioritized visual view of the high-level requirements and schedule - Use it to estimate total # of sprints to calculate cost estimates
Release Plan	- Contains a release schedule for requirements - Use it to estimate total # of sprints to calculate cost estimates
Product Backlog	- A complete list of all currently known product requirements - Use it to prioritize and eliminate features when managing costs

Lecture

04

Wrapping-Up



Lecture
04

Key Project Management Events/Ceremonies

Planning Session



Product Roadmap
Release Planning
Initial Product Backlog
Initial Cost Estimate
User Stories
Story Mapping

Iteration/Sprint Planning



User Stories
Sprint Backlog

Iteration/Sprint Retrospective



Product Backlog Refinement
Product Roadmap Refinement
Release Planning Refinement

Strengths and Weaknesses of Agile Approach

Strengths

Simplicity

Flexibility

Fast delivery and
decision making

Focus on Value Adding

Weaknesses

Lack of Clarity at
beginning

Less predictability

Requires motivated
and committed team

Difficulties managing
large projects

Lecture

04

Key Points

Agile approach does project scope, time, and cost in a simpler manner

Typically, in Agile projects Scope is variable, Time and Cost are fixed

Estimating time is based on story points for each user story, and Velocity determines how many stories should be put in a Sprint

Cost estimating is based on cost/sprint and velocity

All management activities are done in planning session, iteration planning, and iteration retrospective

Preparation for discussion panel:

- Watch the video:
[“Learn Agile Estimating”](#)
- Question: what are the benefits of using Agile estimating techniques?



Lecture

04

For Discussion Panel

Before next lecture:

- For Next Lecture, Watch the Video [” Agile Methodologies - XP | Kanban | Lean | Scrum | Crystal | DSDM | ASD”](#)

Lecture 04

For Next Lecture

- To study/review the whole course
Watch the video:
[“Agile Management | Google Project Management Certificate”](#)





Thank You