



Agile-based Engineering

IT2080 IT Project

B. Sc. Special Honors in Information Technology

Year 2 – Semester 2

Agenda

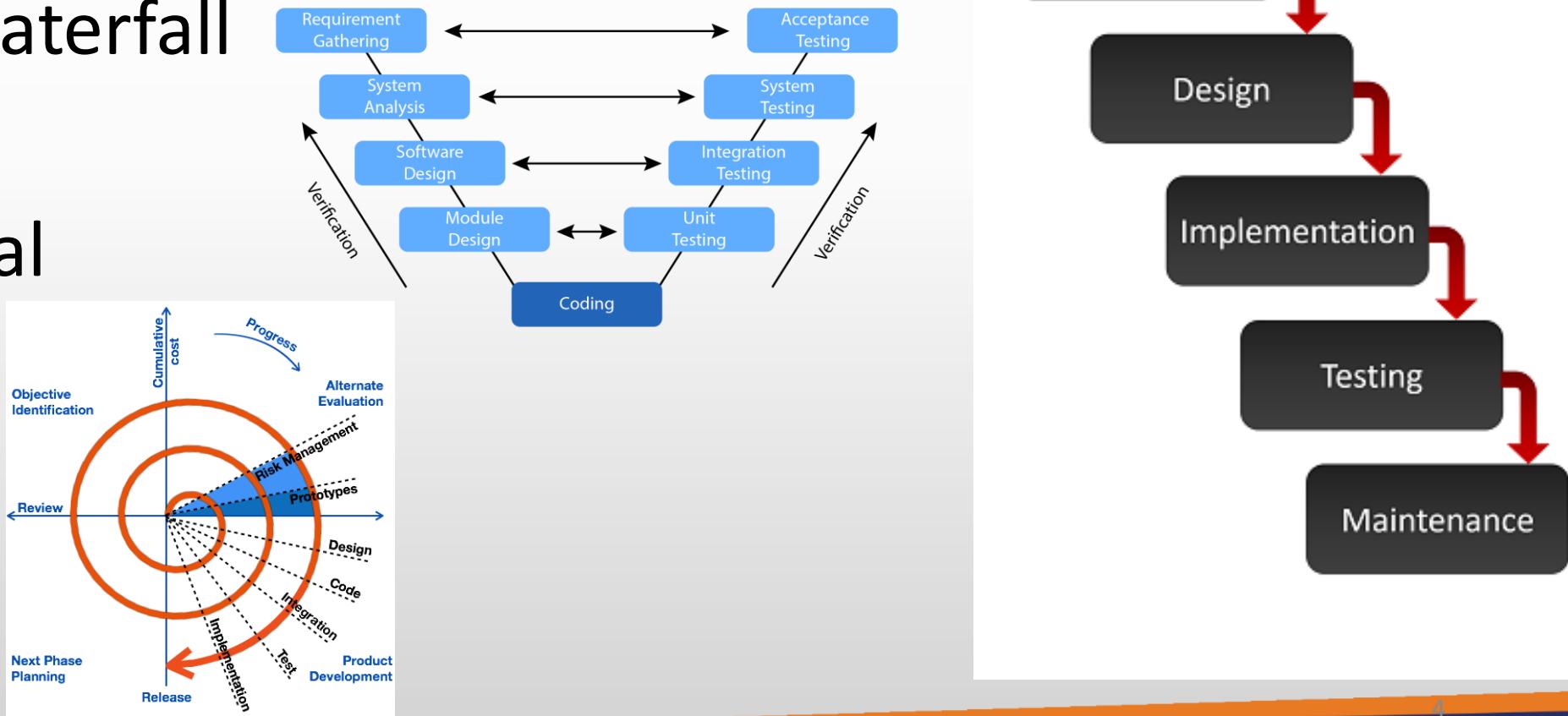
1. Why agile methodologies?
2. Agile approach
3. Agile-based project scope management
4. Agile-based project schedule management
5. Some agile methodologies

1. Why agile methodologies?

Traditional methodologies

Based on the waterfall model

- Iterative waterfall
- V model
- Incremental
- Iterative



Traditional methodologies

- Documentation-based
- Depend on processes and tools
- Client's do not see a product early so, difficult to evaluate
- Engineers and clients may have idea gaps

2. Agile approach

Agile Manifesto

Four Paired Values

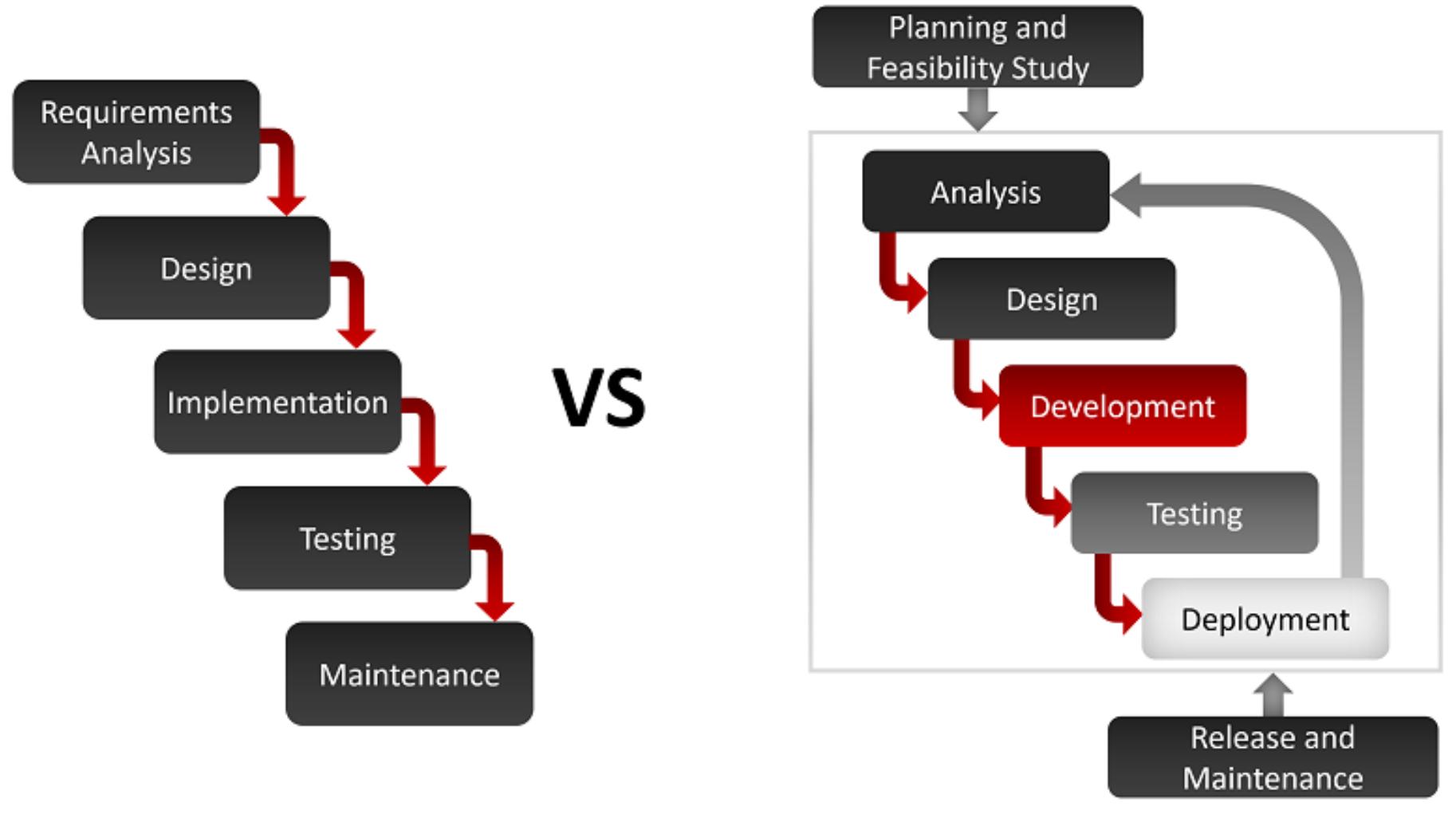
1. **Individuals and interactions** over processes and tools
2. **Working software** over comprehensive documentation
3. **Customer collaboration** over contract negotiation
4. **Responding to change** over following a plan

<https://agilemanifesto.org/>

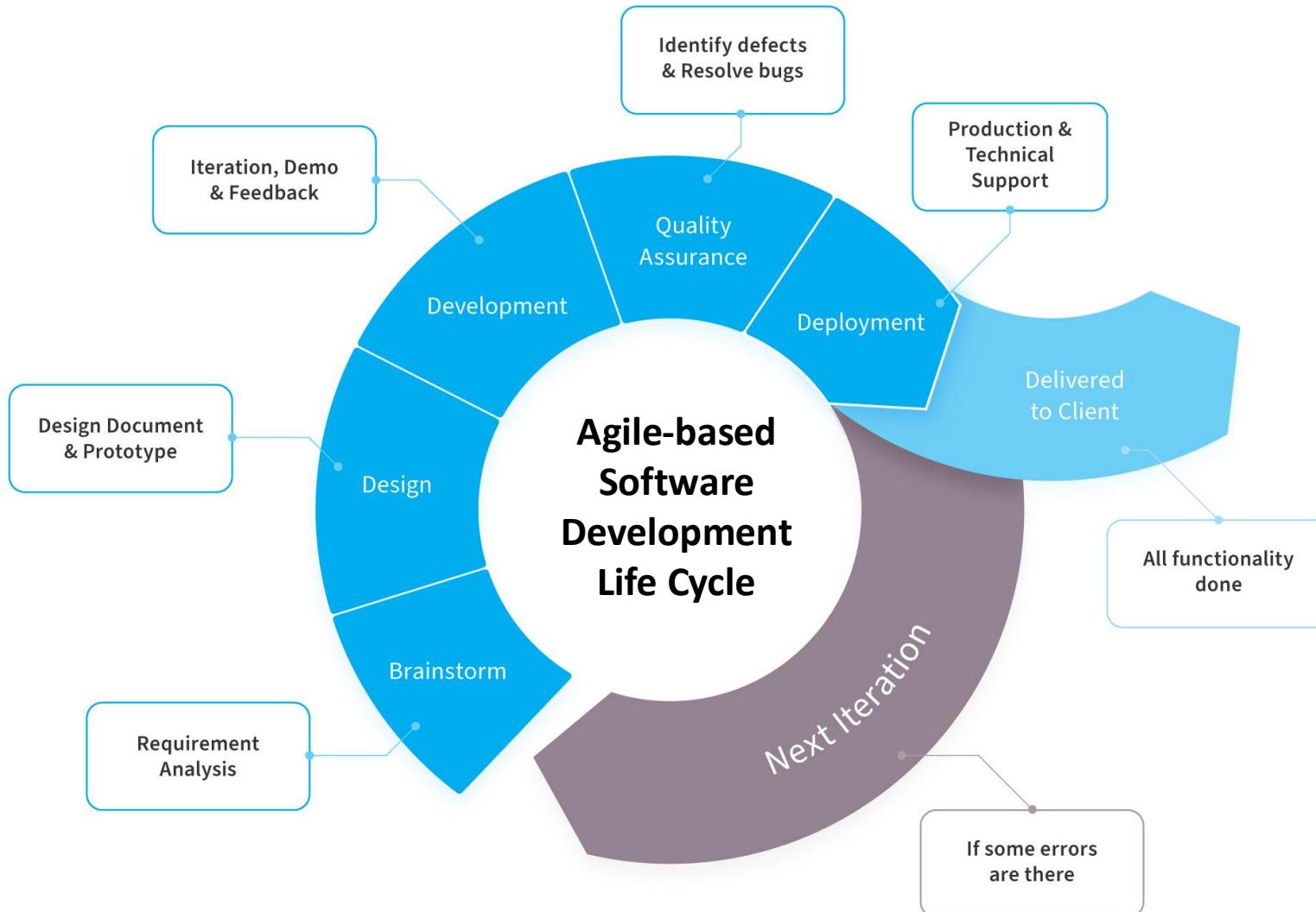
Agile approach over traditional approach

	Predictive	Iterative	Incremental	Agile
Requirements	Fixed	Dynamic	Dynamic	Dynamic
Delivery	Single delivery	Single delivery	Frequent smaller deliveries	Frequent small deliveries (sprints)
Change	Constrained as much as possible	Incorporated at periodic intervals		Incorporated in real-time delivery
Focus	Manage cost	Correctness of solution	Speed	Customer value
Stakeholder involvement	Only at specific intervals or milestones	Regular involvement		Continuously involved
Work	Generally performed once on the project	Repeated until correct	Performed once per increment	Continuously repeated
Best suited for	Well understood projects	Scope determined early but can be modified	Series of iterations that successively add functionality	Rapidly changing environment

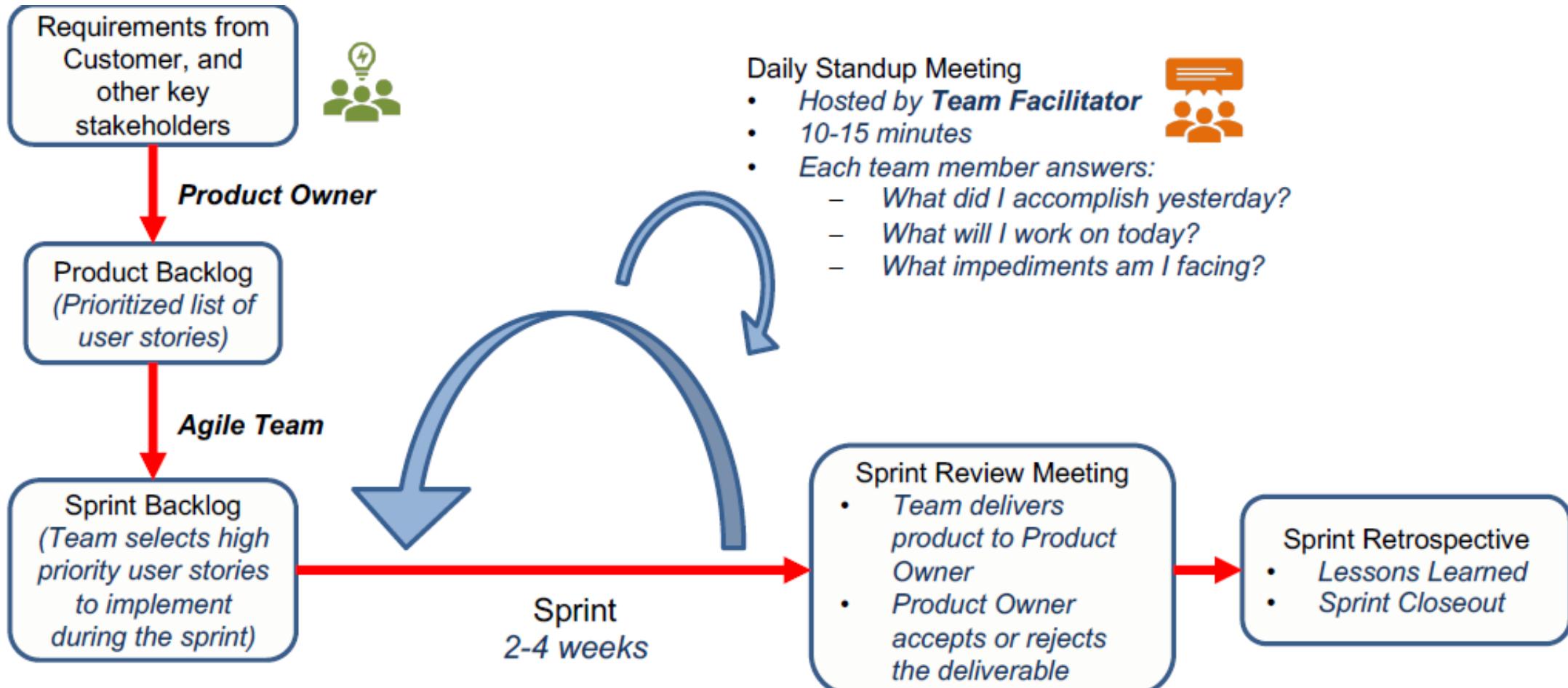
Agile approach over traditional approach



Agile Lifecycle – Iterative and Incremental



Agile Lifecycle – Iterative and Incremental



Agile-based team roles

Agile Team

- *Is self organizing and includes developers, testers, business analysts, designers, etc.*
- *Cross functional with T-shaped skills*



Agile Coach

Image credit: Palto/Shutterstock

- *Mentors and coaches the Agile team members*
- *Optional role (often played by experienced team facilitators)*

Agile-based team roles

Team Facilitator (Scrum Master)

- *Servant Leader who encourages collaboration between team members*
- *Removes impediments and ensures teams have the tools to complete the work*

Product Owner

- *Represents the business (customer)*
- *Prioritizes user stories to create the product backlog*
- *Approves items delivered during the sprint review*

3. Agile-based project scope management

Agile-based project scope management

- Product Backlog
 - *Prioritized list of user stories - by the Product Owner*
 - *Sprint backlog/Iteration backlog created from the Product backlog*
 - *Agile team produces high level relative estimates*
 - *Backlog items can be reprioritized at any time (backlog grooming)*
- User Stories

“As a <Role>, I want <Functionality>, so that <Business benefit>”

For example:

“As a call center analyst, I want to search clients by first and last name so I can look up their records quicker.”

Acceptance Criteria

IN ORDER TO save time when banking and
not have to wait in line or opening hours

AS Customer of the Bank

I WANT to be able to manage my account at
any time (day or night)

ACCEPTANCE CRITERIA

I want to be able to find out the balance of my account
I want to be able to know all recent transactions
etc, etc

Acceptance Criteria example

User story: As a customer I need to place an order so that I can get my food delivered

Acceptance criteria

1. Functional:

Can I save my order and come back to it later?

Can I change my order before I make the payment?

Can I see the running total cost of what I have chosen?

2. Non-functional: availability:

Can I place an order at any time (24 hours per day or 24/7/365)?

Can I view the order at any time (24 hours per day or 24/7/365) up to and including delivery?

Non-functional: security:

Are other customers prevented from viewing my order?

Agile-based project scope management

Definition of Done

- Criteria to determine if the work is complete
- Agreed by Product Owner, Agile team and Team Facilitator during Release Planning

Definition of Ready

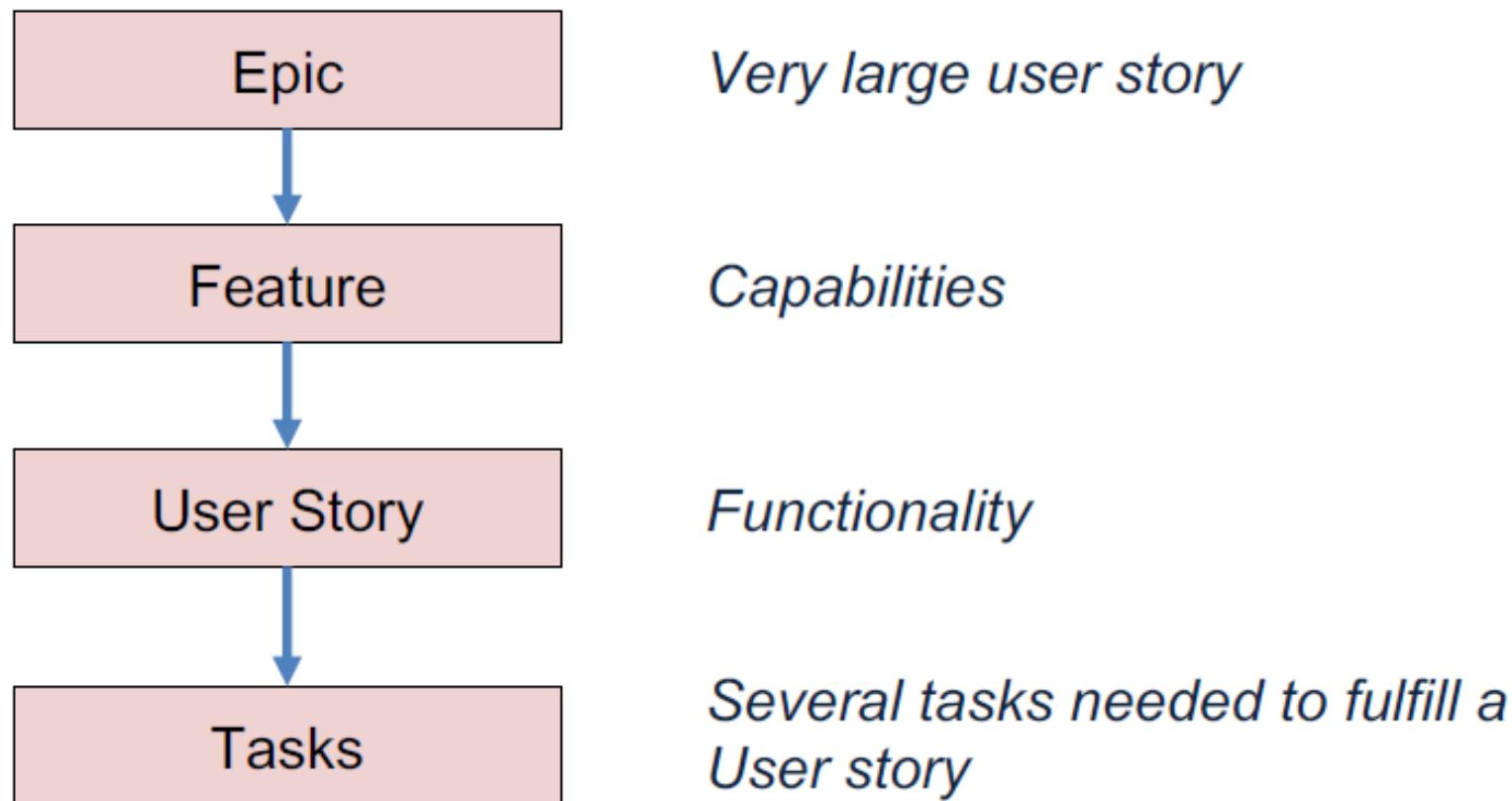
- Checklist to determine that all information is available for the team to start working on the user story
- User stories must be immediately actionable

4. Agile-based project schedule management

Agile-based project schedule management

- Agile uses a Product Roadmap (not a schedule)
- Iterative Scheduling with a Product Backlog
 - *User stories are prioritized based on priority and time*
 - *New stories can be added to the backlog*
 - *Does not always work well if there are complex dependency relationships*
- On-demand Scheduling
 - *Team members “pull” work from a queue*
 - *Based on Kanban and Lean methodologies*
 - *Works best when activities can be divided into equal sizes*
 - *Does not always work well if there are complex dependency relationships*

Agile-based project schedule management



Estimating Methods, Prioritization, and Consensus Gathering

Estimating Methods – Agile

- T-shirt Sizing
 - *Follows sizes of t-shirts such as: XS, S, M, L, XL*
- Story Pointing
 - *Uses the Fibonacci Sequence of numbers*
 - *1, 2, 3, 5, 8, 13, 21, 34, 55*
- Planning Poker
 - *Uses a deck of cards with a modified Fibonacci sequence*

Estimating Methods, Prioritization, and Consensus Gathering

Prioritization (Agile)

- MoSCoW Analysis
 - *Must Have*
 - *Should Have*
 - *Could Have*
 - *Won't Have*
- Kano Model
 - *Satisfies vs. delights vs. dissatisfies vs. indifferent*
- Paired Comparison Analysis
 - *Prioritization of successive pairs*
- 100 Point Method
 - *Prioritization by spreading points across user stories*

5. Some agile methodologies

Some agile methodologies

Kanban

- Kanban is a popular framework used to implement agile and DevOps software development.
- It requires real-time communication of capacity and full transparency of work.
- Work items are represented visually on a Kanban board, allowing team members to see the state of every piece of work at any time.

<https://youtu.be/iVaFVa7HYj4>

Some agile methodologies

Scrum

- Scrum is a framework that helps teams work together.
- Scrum encourages teams to learn through experiences, self-organize while working on a problem, and reflect on their wins and losses to continuously improve.

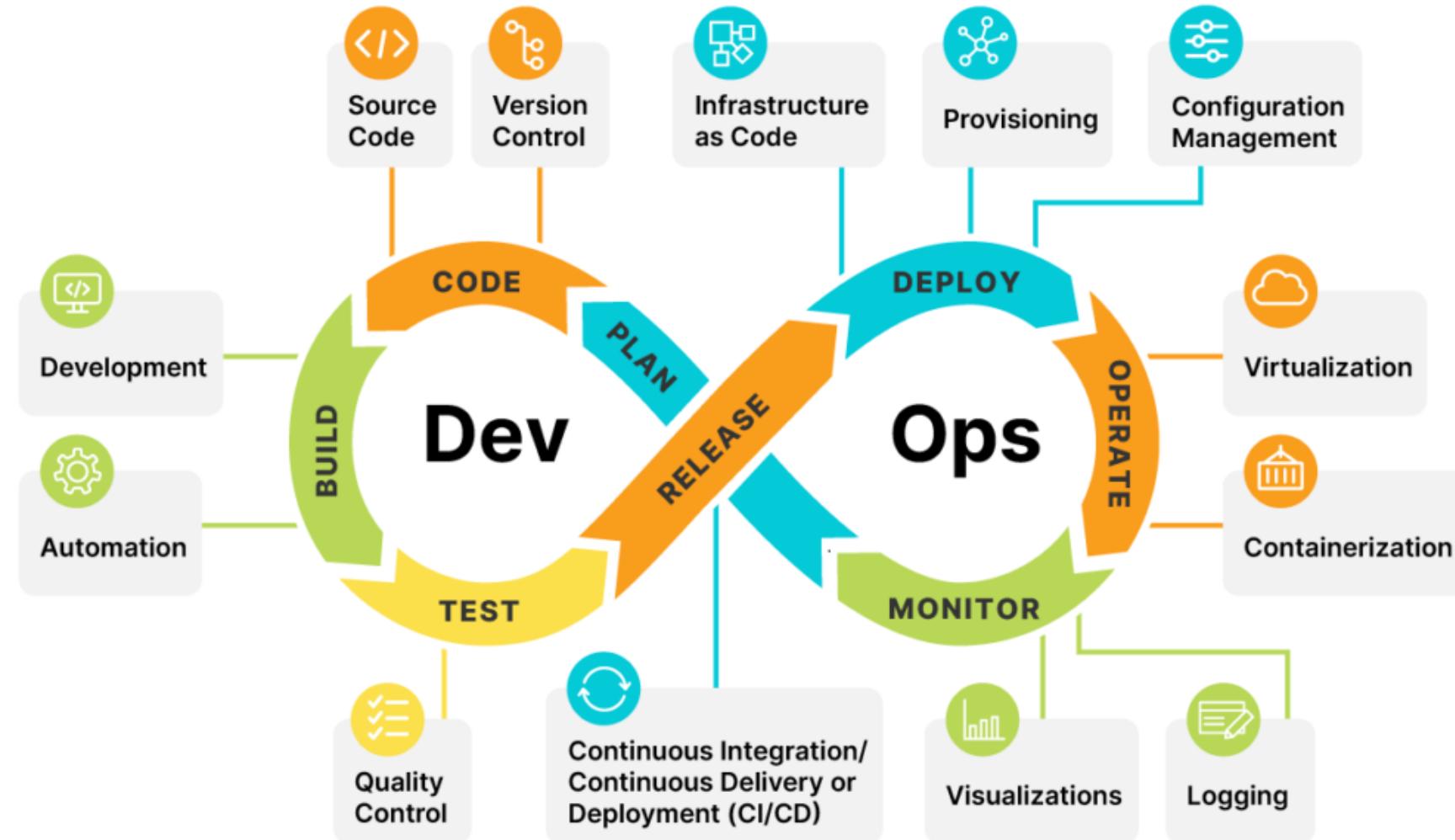
<https://youtu.be/b02ZkndLk1Y>

Some agile methodologies

- “Kanban vs. scrum” is a discussion about two different strategies for implementing an agile development or project management system.
- Kanban methodologies are continuous and more fluid, whereas scrum is based on short, structured work sprints.

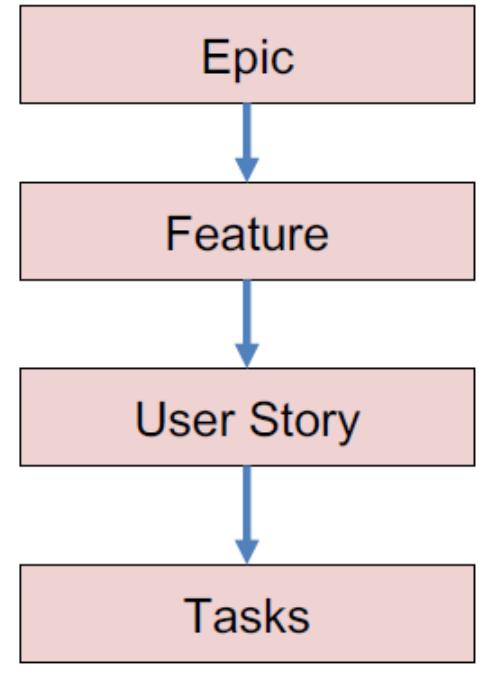
Some agile methodologies

DevOps



Summary

1. Why agile methodologies?
2. Agile approach
3. Agile-based project **scope management**
4. Agile-based project **schedule management**
5. Some agile methodologies



<https://www.coursera.org/articles/what-is-agile-a-beginners-guide>

Activity 3

The objective of this activity is to plan, prioritize, and manage the development process by employing agile methodologies.

1. Prepare a Product backlog for your project
2. Assign priority for each user story/ or requirement
3. Add relative estimates for the user stories using agile estimation method.
4. Based on the priority and estimates given, separate them to 3 sprints(duration of each sprint is two weeks). Each sprint should have a product as an outcome. The final sprint (3rd) should have the fully developed product
5. Create the Kanban board (task board) for ITP project using a suitable project management tool (Ex: Trello. Jira or any other). Ensure following steps in creating the Kanban board.
Add all user stories in product backlog to the Kanban board and all tasks identified for each user story.
You may attach a screenshot for the submission
6. Provide a sprint plan showing the tasks allocated for each member.

