

# AGILE-PARADIGM SHIFT IN SDLC

## Session - 7

### Agile Software Development - II



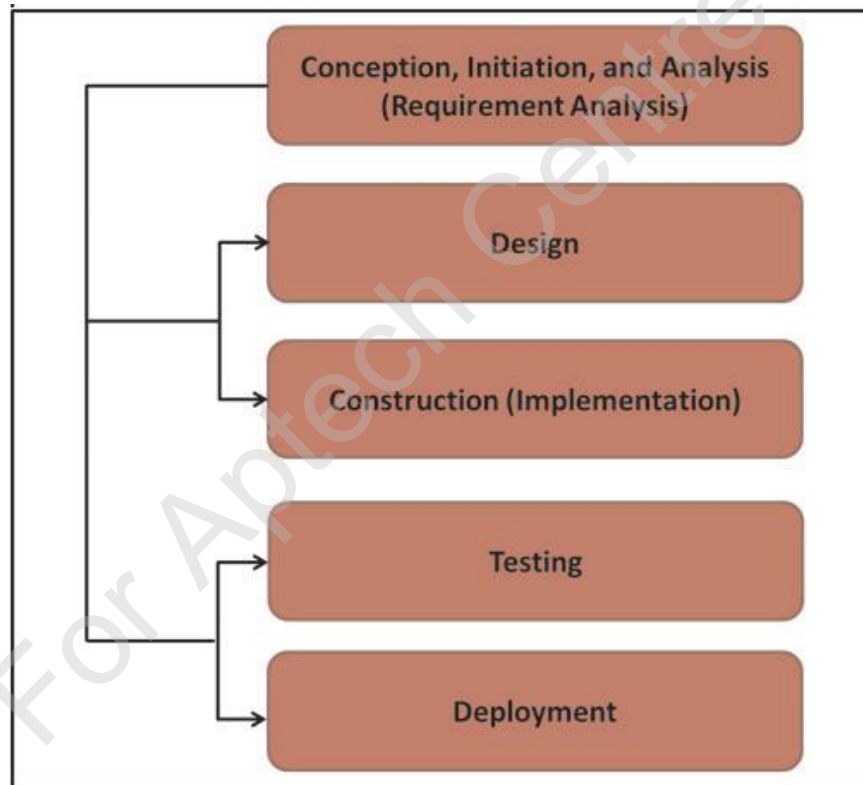
# Objectives

- ◆ Describe the phases in Agile development
- ◆ Describe Agile artifacts
- ◆ List the advantage and disadvantages of Agile software development
- ◆ Describe the various Agile tools and mechanisms

For Aptech Centre Use Only

# Phases in Agile Development [1-4]

- ◆ Similar to the traditional process model such as the waterfall model, the Agile model also has several phases.
- ◆ The key difference is that all the Agile phases incorporate collaboration, iteration, and an incremental approach.
- ◆ Figure represents an overview of the phases in Agile development.



# Phases in Agile Development [2-4]

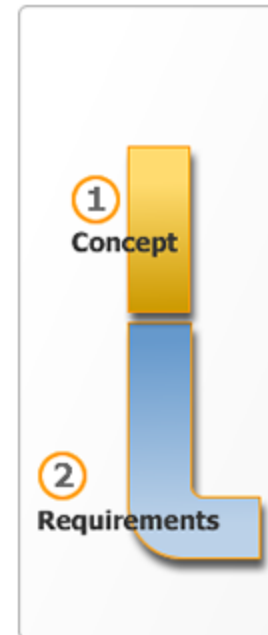
## Conception Phase

- ◆ Referred to as Iteration -1.
- ◆ Helps to define the problem or business opportunity, that leads to identify and assess a feasible strategy for the project.



## Requirement Phase

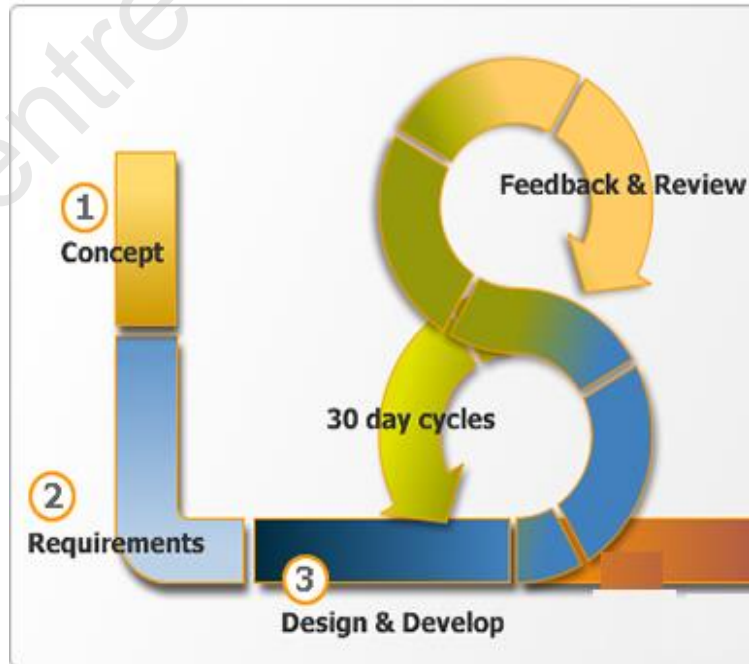
- ◆ Takes place during the first week.
- ◆ Referred to as iteration 0.
- ◆ Project is initiated through a feasibility study.
- ◆ Work with stakeholders to determine the initial requirement.
- ◆ Involves requirement analysis



# Phases in Agile Development [3-4]

## Design and Development Phase

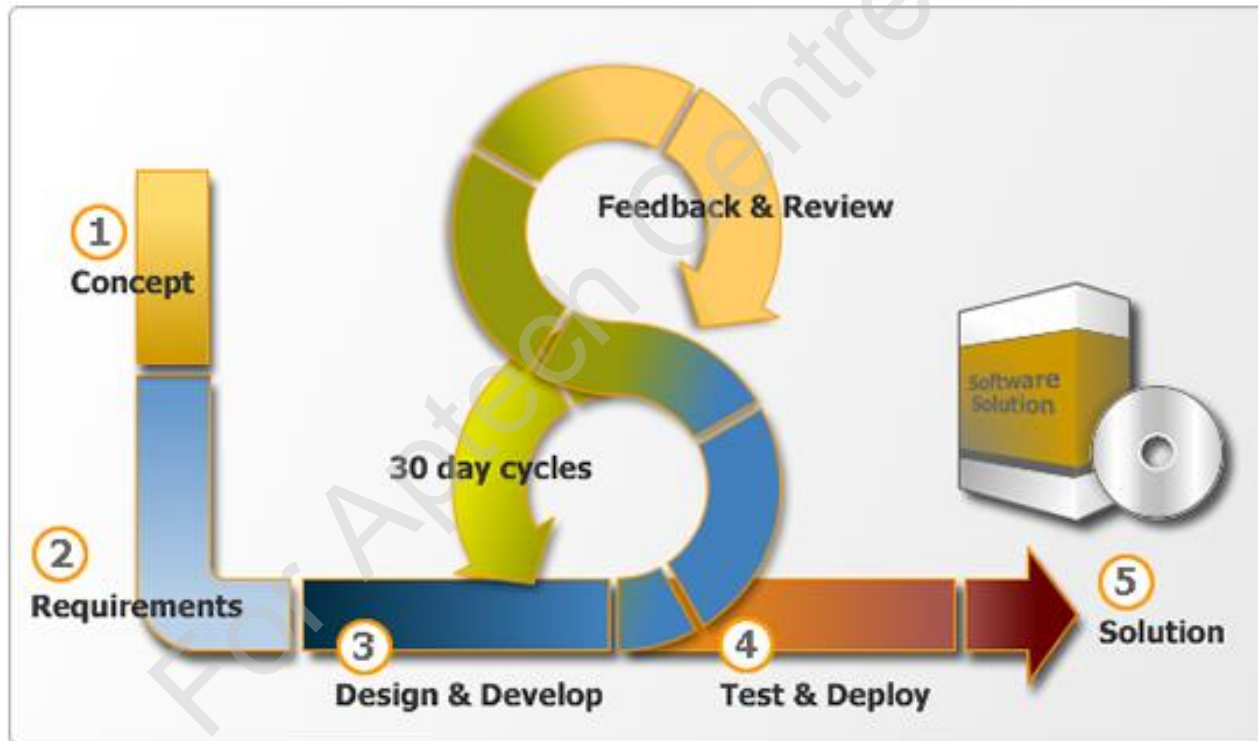
- ◆ **Design** phase involves creating design strategies to model the system.
- ◆ **Implementation** phase begins the project's actual development of the project.



# Phases in Agile Development [4-4]

## Testing and Deployment Phase

- ◆ **Testing** will evaluate and test the completed project on various aspects and factors.
- ◆ **Deployment** is the final stage of launching the project.



# Agile Artifacts [1-3]

- ◆ **Artifacts** are created for a specific purpose and used accordingly to develop software.
- ◆ To measure a project's progress, Agile project teams often use the following main artifacts or deliverables:
  - ◆ Product vision statement
  - ◆ Product backlog
  - ◆ Product roadmap
  - ◆ Release plan
  - ◆ Sprint backlog
  - ◆ Increment

# Agile Artifacts [2-3]

## Product vision statement

- Comprises a short overview statement and a vision statement to express the goals or outcomes of the product.

## Product backlog

- Comprises a list of tasks for the project, ordered by priority.

## Product roadmap

- Comprises a roadmap for the product requirements, with an approximate time frame for the development of those requirements.

## Release plan

- Comprises an evolving timetable that describes the features to be delivered in upcoming releases.

## Sprint backlog

- Comprises a list of tasks to be completed within the current sprint ranked in the desired order of implementation.

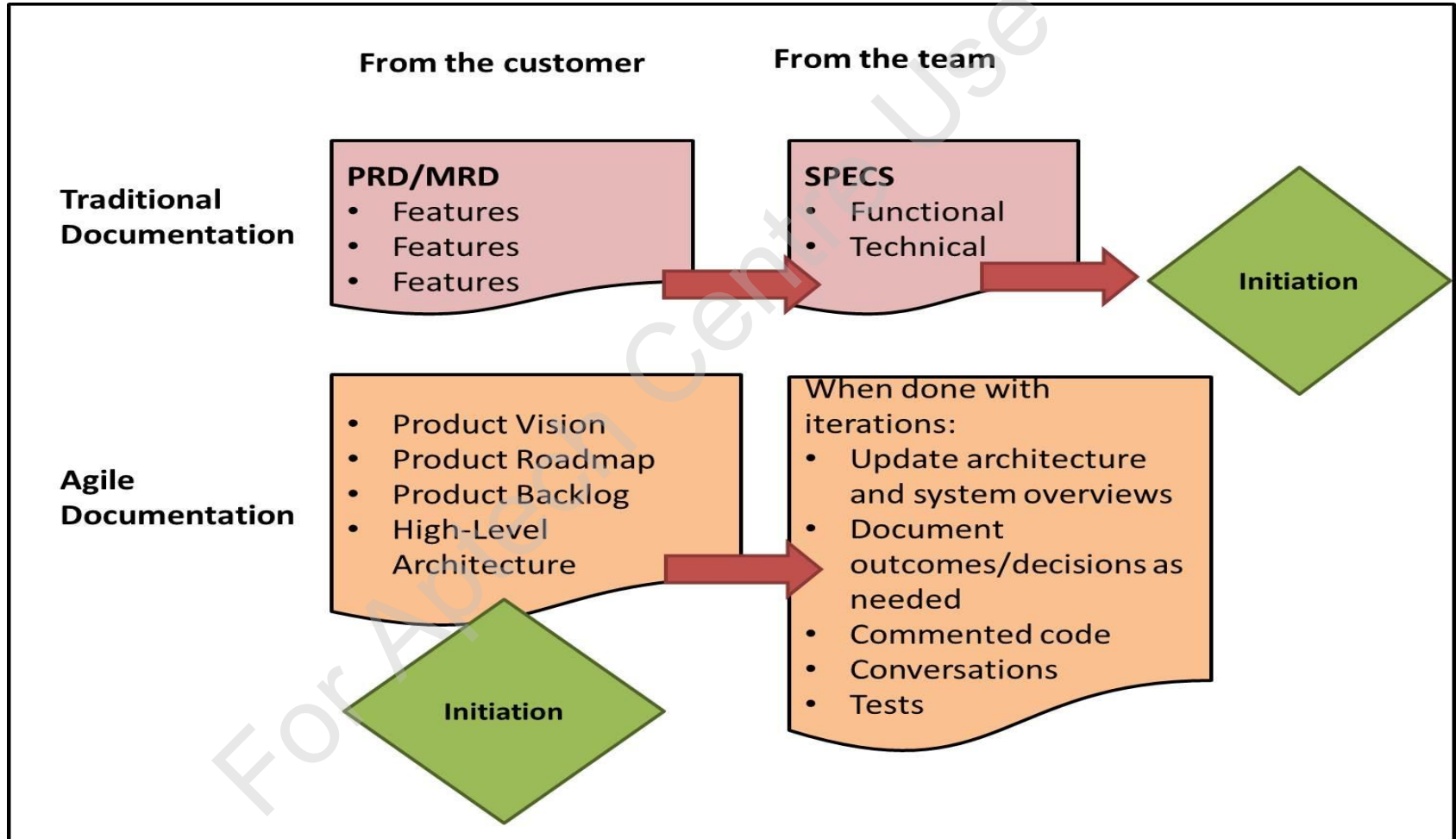
## Increment

- Indicates the product functionality at the end of each sprint.



# Agile Artifacts [3-3]

- Figure shows the differences between traditional and Agile documentation.



# Agile Advantages and Disadvantages

Advantages	Disadvantages
Very realistic in its approach.	Not suitable for handling complex dependencies.
Promotes teamwork and cross training.	Requires skilled manpower.
Functionality can be developed quickly and demonstrated.	Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
Minimum resource requirements.	High individual dependency leads to minimum documentation.
Applicable for fixed or changing requirements.	Heavily dependent on customer interaction for collecting changing requirements.
Delivers partial working solutions early in the cycle.	Partial solutions may not cover all the requirements.
Suitable for regularly changing environments.	Requires strong change management system.
Minimal rules.	Adhoc rules may have to be made.
Development and delivery happens in parallel within an overall planned context.	Invariably requires an overall plan.
Easy to manage.	Risk of sustainability, maintainability, and extensibility.

# Agile Tools and Mechanisms [1-6]

- ◆ The major Agile tools and mechanisms used today are as follows:

**Velocity**

**Burndown and Burnup charts**

**Informative Workspace**

**Big Visible Charts**

**Hand-drawn Charts**

**Process Improvement Charts**

**Task Boards**

# Agile Tools and Mechanisms [2-6]

## Velocity

- ◆ Is measured by the quantity of work that the team members together can accomplish in one iteration.
- ◆ Is used to measure the time taken for a team to deliver upcoming outcomes based on its prior performance.
- ◆ Is the data used to calculate a team's velocity is depicted on the project burndown and burnup charts.

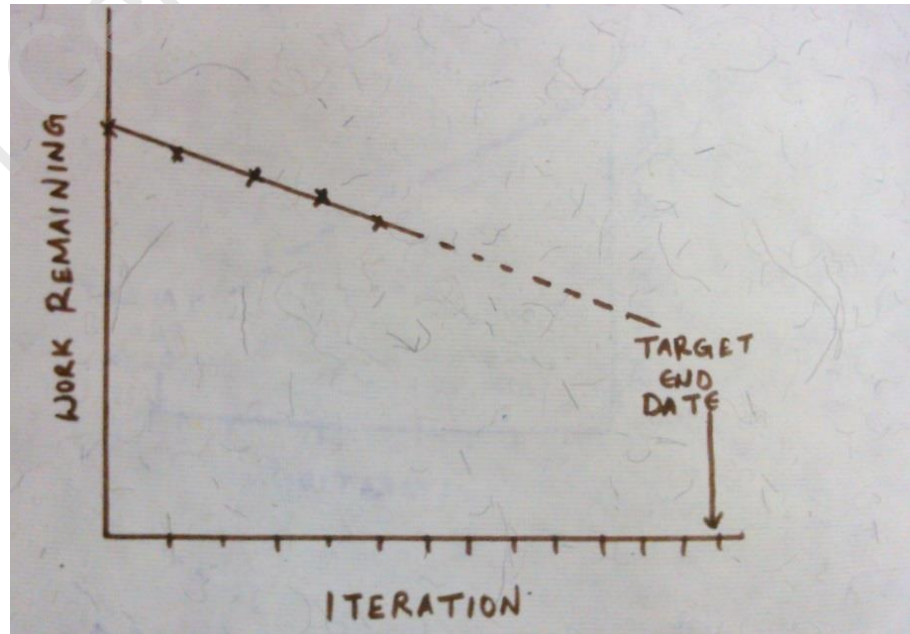
## Burndown and Burnup Charts

- ◆ Big visible charts are big charts placed near the Agile team workspaces that convey the progress of the project team in various ways.
- ◆ Charts aid the team in working effectively towards the overall progress in developing the software.

# Agile Tools and Mechanisms [3-6]

## Burndown Charts

- ◆ A burndown chart is one of the most important tracking tools for Agile projects.
- ◆ It tracks the work remaining on the project and the possibility of the deadline being met.
- ◆ The vertical axis measures the work remaining and the horizontal axis marks the iterations.



# Agile Tools and Mechanisms [4-6]

## Burnup Charts

- ◆ It tracks the amount of work done.
- ◆ It shows the quantity of work in the whole project.
- ◆ It shows changing goal or scope of the project.
  - ◆ The bottom line represents the work done and the top line, the total scope.
  - ◆ Dotted lines show projections.

## Process Improvement Charts

- ◆ Process improvement charts are a type of big visible charts.
- ◆ It measures specific issues that the team would like to improve upon.
- ◆ These charts stay only as long as they are necessary.

# Agile Tools and Mechanisms [5-6]

## Other Charts

- ◆ The iteration and release planning boards are other examples of big visible charts.
- ◆ The team calendar is another useful status chart that display important dates and iteration numbers.

## Informative Workspace

- ◆ An informative workspace is not a tool, but rather a practice.
- ◆ It is a workspace that has feedback mechanisms incorporated to help the people who are working in it.

# Agile Tools and Mechanisms [6-6]

## Hand-drawn Charts

- ◆ As the information in the charts constantly change, it is easier to use flip charts and whiteboards to draw a chart than creating one on a computer.

## Task Board

- ◆ A task board represents at a minimum, three columns, '**to-do**', '**in-progress**', and '**done**', and is used for organizing a team's work.
- ◆ It mostly displays the work to be performed during the current iteration.



# Summary

- ◆ In Agile development, all phases incorporate collaboration, iteration, and an incremental approach.
- ◆ New features identified in the course of a life cycle of an Agile project are planned across sprints.
- ◆ A sprint is the most basic unit of development and is defined as a fixed period of time during which a specific task has to be completed and made ready for review.
- ◆ The duration of each sprint depends on the feature to be developed. Each sprint goes through all the phases.
- ◆ In Agile development, each operational team (such as business analysts, technical architects, developers, testers, and so on) must participate collaboratively.
- ◆ Some of the artifacts of Agile projects include Product vision statement, Product backlog, Product roadmap, and so on.
- ◆ The major tools and mechanisms used in Agile software development are Velocity, Informative Workspace, Big Visible Charts, Hand-Drawn Charts, Task Boards, and so on.