FINAL DRAFT International Standard

ISO/IEC FDIS 33202

Software and systems engineering — Core agile practices

ISO/IEC JTC **1**/SC **7**

Secretariat: **BIS**

Voting begins on:
2024-05-14

Voting terminates on:
2024-07-09

Reference number
ISO/IEC FDIS 33202:2024(en)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

Many organizations in the software development industry are adopting agile life cycle to adapt to rapidly changing and evolving marketplaces. “Agile” is derived from the Agile Manifesto[10], which was designed to ‘uncover better ways of developing software by doing it and helping others do it’. It emerged as an alternative to documentation-driven, heavyweight software development processes. The Agile Manifesto[10] and related values and principles (Annex C) have served as reference for agile practices across many organizations.

Agile development is relative to how each organization interprets the agile values and principles (See Annex C). The agile practices discussed in this document offer a lightweight, adaptive and collaborative development approach based on empiricism with the focus on rapid business-value delivery.

Agile standardization refers to adopting agile practices and concepts across an organization or project for software suppliers, software acquirers, software service providers, large delivery and maintenance projects and system integration engagements having software components.

Standardization of agile practices is useful for the following reasons:

a) for suppliers: it provides a basis for delivering meaningful products and services;

b) for acquirers: it provides a basis to differentiate between different suppliers and provides guidance on their role in the development.

The purpose of this document is to catalogue the core agile practices (see Annex A) and concepts demonstrated by organizations or projects. These practices and concepts enable the application of agile in acquisition, supply, development, operation, maintenance and disposal of software systems, products and services.

This document can be used by acquirers, suppliers, developers, integrators, operators, maintainers, managers, quality assurers and users of software systems, products and services, for a range of situations and contexts.

Software and systems engineering — Core agile practices

1 Scope

This document defines a set of core practices and concepts that have wide acceptance in organizations and industries using agile practices and concepts. This document defines a set of core practices that are present in agile methodologies.

The practices and concepts defined in this document are applicable to a single agile team, as well as to multiple agile teams. These practices and concepts are applicable throughout the life cycle of software systems, products and services.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
acceptance criteria
criteria that a system or component must satisfy in order to be accepted by a user, customer, or other authorized entity

[SOURCE: ISO/IEC/IEEE 24765:2017, 3.32, definition 1]

3.2
agile
approach to development, delivery and maintenance of products and services by enabling rapid response to feedback

Note 1 to entry: The need specification for the software under development is elaborated with specific requirements only when the work is started. This lean principle is meant to avoid waste of work and to provide *agile team* (3.6) with means of prioritizing their work.

3.3
agile concept
fundamental building blocks of *agile* (3.2)

3.4
agile development
software development approach based on iterative development, frequent inspection and adaptation, and incremental deliveries, in which requirements and solutions evolve through collaboration in cross-functional teams and through continuous stakeholder feedback

[SOURCE: ISO/IEC/IEEE 24765:2017, 3.119]

3.5
agile environment
organizational culture, infrastructure, and methodologies that support *agile development* (3.4)

[SOURCE: ISO/IEC/IEEE 26515:2018, 3.2]

3.6
agile team
organization or team using *agile development* (3.4) methods and approaches

[SOURCE: ISO/IEC/IEEE 26515:2018, 3.3, modified — Note 1 to entry has been removed.]

3.7
architecture
fundamental concepts or properties of an entity in its environment and governing principles for the realization and evolution of this entity and its related life cycle *processes* (3.28)

[SOURCE: ISO/IEC/IEEE 42010:2022, 3.2]

3.8
architecture description
work product used to express an *architecture* (3.7)

[SOURCE: ISO/IEC/IEEE 42010:2022, 3.3, modified — The abbreviated term and notes to entry have been removed.]

3.9
backlog
collection of features or stories of both functional and non-functional requirements that are typically sorted in an order based on value priority

Note 1 to entry: Backlog traditionally means things that are not completed.

3.10
continuous delivery
software engineering *practices* (3.20) that allow for frequent releases of new systems (including software) to staging or various test environments through the use of automated tools

[SOURCE: ISO/IEC/IEEE 32675:2022, 3.1]

3.11
continuous deployment
automated *process* (3.28) of deploying changes to production by verifying intended features and validations to reduce risk

[SOURCE: ISO/IEC/IEEE 32675:2022, 3.1]

3.12
continuous improvement
ongoing cycle of identification, implementation and evaluation of changes to how *practices* (3.20) are used that strengthen the ability of an organization or project to meet its objectives

3.13
definition of done
DoD
statement on the required quality attributes that work meets before the work completes a specified life cycle activity or task and is ready for use

3.14
definition of ready
statement on the required quality attributes a *backlog* (3.9) item or other work product meets before it starts a specified activity or task

3.15
gemba
observation of work *processes* (3.28) actually in use

3.16
improvement backlog
intentional, emergent, ordered list of items of how to make improvements

3.17
increment
tested, deliverable version of a software product that provides new or modified capabilities

3.18
iteration
short time frame in which a set of software features is developed, leading to a working product that can be demonstrated to stakeholders

Note 1 to entry: Some *agile* (3.2) methodologies are not based on iterations.

[SOURCE: ISO/IEC/IEEE 26515:2018, 3.10, modified — The original note 1 to entry has been removed; note 2 to entry has been renumbered as note 1 to entry.]

3.19
iteration backlog
subset of a *backlog* (3.9) chosen for inclusion in the current *iteration* (3.18)

3.20
practice
way of consistently performing activities that contributes to achieving a specific purpose

3.21
product backlog
prioritized listing of product requirements

3.22
product owner
designated stakeholder accountable for defining and accepting outcomes of the work and managing *backlog* (3.9), while aligning with the stakeholder needs

3.23
release backlog
subset of *product backlog* (3.21) planned for the next release

3.24
statement of work
statement of the expected outcomes and outline of the work required to achieve the outcomes

Note 1 to entry: Statement of work is unique and distinct for each project and often provides a measurable *definition of done* (3.13). It is usually a contractual document prepared during project initiation and planning that describes what outcomes are expected and outlines the work required to achieve these outcomes.

3.25
testing pyramid
graduated series of tests which includes many simple and automated tests (unit tests) with less frequent integration tests and few lengthy end-to-end or manual tests

3.26
user story
brief description of desired functionality

Note 1 to entry: User story can describe the stakeholder roles, goals, benefits, and motivation

Note 2 to entry: Work in backlog can be referred in other ways like epics, features

3.27
working agreement
statement of the expected working relationship between two or more parties in an agile environment

EXAMPLE Between members of an *agile team* (3.6), between acquirer and supplier, or between multiple agile teams.

3.28
process
set of interrelated or interacting activities that transforms inputs into outputs

[SOURCE: ISO/IEC/IEEE 12207:2017, 3.1.33]

4 Conformance

4.1 Intended usage

The requirements in this document are contained in Clause 6. This document provides requirements for agile practices suitable for usage by an organization or project. Particular projects or organizations may need only some of the practices provided by this document. Therefore, implementation of this document typically involves selecting and declaring a set of practices suitable to the organization or project. Claiming full conformance asserts that all of the required outcomes of the declared set of practices are achieved.

NOTE An organization (e.g. national or industrial association or company) imposing this document as a condition of trade can specify and make public the minimum set of required practices, and outcomes which constitute suppliers' compliance with the conditions of trade.

4.2 Full conformance

A claim of full conformance declares the set of practices for which conformance is claimed. Full conformance to outcomes is achieved by demonstrating that all of the outcomes of the declared set of practices have been achieved.

5 Conceptual foundations

5.1 Practices and processes

Processes are often considered as a high-level view of the organization or project's activities. These activities require the allocation of resources such as people, tools and materials. Every process has a clearly identified input and output. Such inputs and intended outputs can be tangible or intangible.

Practices are pragmatic actions that are repeatable, regular and recognizable in a context. They describe what really happens rather than what is expected to happen. A practice can be considered as a sound way of working on a problem so that a certain specified quality results in the solution. A practice is a way of consistently performing activities that contributes to achieving a specific purpose (this document). The specifics of the practice are left to the individual or team who make the practice work based on their knowledge and skills.

Figure 1 depicts key concepts pertaining to practices and processes.