

SDEVEN Software Development & Engineering Methodology

Version: 7.0.12

Last update: 230805

SDEVEN Software Development & Engineering Methodology

Table of Content

- [SDEVEN Software Development & Engineering Methodology](#)
 - [Overview](#)
 - [SDEVEN procedures](#)
 - [SDEVEN appendices](#)
 - [Books and other reference materials](#)
 - [FAQ. Frequently Asked Questions](#)

Overview

The RENware Software Development Methodology (**SDEVEN**) is the basic set of rules applicable in software development process and other related processes.

Methodology itself is mandatory in the mean that is ***always applicable ref to software production in company***, regardless to project, software type, team members or composition, project or system classification, location, etc.

The methodology is "flexible enough" to allow changes and liberty for innovation and creativity but "rigid" to practices that may be at law limits or out of law, team members cooperation and collaboration, equity, respect for anyone work and contribution, respect for intellectual property and integrity, etc. Anyone can manifest its own creativity and innovation as long as this is made known and accepted and not induce negative effects on projects and its members*.

SDEVEN procedures

SDEVEN methodology consists of and includes the following sections (procedures):

- **Administrative** section [SDEVEN.10-ADM](#)

This section address important aspects related to administrative procedures, who is expected to do some things and how to handle them.

- **Practices and technical issues** section [SDEVEN.20-RENPR](#)

In this section will be found some of the most important practices and advices regarding the software development.

- **Versioning** section [SDEVEN.30-RENVER](#)

This section treat the *git* repository for software in particular how to use version codes and tags.

- **Branches** section [SDEVEN.40-BRAN](#)

This section treat the *git* repository for software in particular how to use branches in order to avoid misunderstandings and to have your work as clean as possible in order to be shared and used by the other team members and colleagues.

- **Operational Meetings** section [SDEVEN.45-OPME](#)

This procedure will treat software development production and planning meetings.

- **Change Management** section [SDEVEN.50-CHGMNG](#)

This section is about how changes should be handled and treated.

- **System dev logging changes / updates** section [SDEVEN.55-TRACE](#)

This section is about "*in development system*" tracing: issues, features, fixes, and generally speaking all **TRACING & LOGGING** of intentions, changes, proposals, and so on.

- **Release management** section [SDEVEN.60-RELM](#)

This section is about the release process, how should be handled and what are the expectations regarding software.

- **Project structure** section [SDEVEN.62-PSTR](#)

This procedure contains usual project structure.

- **Design approaches and their review and change** section [SDEVEN.65-DEREV](#)

This procedure is about design review before being "published" as initial design or after some major | important | high impact changes.

- **Code review** section [SDEVEN.68-COREV](#)

This procedure is about standard code review that should be done before any function, library, package, module, etc beta. The objective, regardless simply conformity, is to "see" potential generalizations and reusability portions.

- **Licenses & products / models** section [SDEVEN.70-LIP](#)

This section is aal brief about licensing process, products, models and related concepts.

- **Code of Conduct. Ethics, Professional and Legal Issues** section [SDEVEN.75-CCEP](#)

This section is about general conduit (general aspects) but also about ethic principles in software development.

- **Conventions & principles** section [SDEVEN.80-COPRI](#)

This section is about terminology and "conventions culture" in order to share a common language and understanding with other team members and colleagues.

- **Phases, Processes and Deliverables** section [SDEVEN.90-RELBLU](#)

This section describe the whole process and a delivery document is proposed.

**Fundamental (basic) methodology phases**

- **(100-ANA) Analysis**
- **(800-SWD) Development**
- **(900-OPS) Operations**

A comprehensive template as suggested structure and content can be [found here](#).

- **Confidentiality & Classification** section [SDEVEN.95-COCL](#)

This section is about projects confidentiality with particular aspects to software development.

SDEVEN appendices

The appendices contains mainly forms, templates, cheat sheets and other helpers. These are:

- [Appendix \(A\) Sematic Versioning \(SEMVER\)](#)
- [Appendix \(B\) System Design Document Template \(810 DSGN\)](#)
- [Appendix \(C\) Status Report Template \(STATUSR\)](#)
- [Appendix \(D\) ROADMAP Template \(RMAP\)](#)
- [Appendix \(E\) Release note template \(RELNOTE\)](#)
- [Appendix \(Q\) Frequently Asked Questions \(FAQ\)](#)

Books and other reference materials

- [Manifesto for Agile Software Development](#)
- [Scrum Guides](#)
- [Creative Commons](#)
- [AGPLv3 GNU Affero](#)

FAQ. Frequently Asked Questions

A list with most common frequent asked questions can be found [here](#).

Last update: August 5, 2023