SDEVEN Software Development & Engineering Methodology

Version: 7.0.6

Release date: 230715

System development and changes logging (SDEVEN.55-TRACE)

Table of Content

Preamble

This procedure is about *in development system* logging and trace issues, features, fixes, etc, generally speaking all "actions" like intentions, changes, proposals, and so on.

Common files used to keep tracking

The following files are mostly present in development projects:

- **CHANGELOG** file - this keep record of all things done in development process - all changes or new things happened
- **ROADMAP** file - this keep record of all things approved on project development roadmap (see *NOTE 1: Files with commercial impact*) - a template document can be found in file Appendix_D_ROADMAP_template.md
- **RELNOTE...version...** file(s) - this keep the record of things already done in any public released version, RELNOTE name is an acronym for "REALEASE NOTES" (see *NOTE 1: Files with commercial impact*)

Outside of these files, projects can have some files dedicated to project management domain, the most usual files "seen by developers" being **Status_report...** (see *NOTE 1: Files with commercial impact*).

> 📝 **NOTE 1: Files with commercial impact**
>
> `ROADMAP`, `RELNOTE...`, `Status_report...` files have commercial impact and could be used by other persons from commercial departments so they should follow just a minimum strictness regarding used language

Taxonomy

Speaking about current type of `situations` that must be traced, basically are: [](

) * **to be done category** representing those things that should be made in a short term - these will be marked with `#TODO` text to be recognized by editing platforms (most of IDEs platform have extensions for that) and visually highlight them

- **bugs or "problems" category** representing those things that create any kind of problems and should be fixed - these will be marked with `#FIXME` text to be recognized by editing platforms (most of IDEs platform have extensions for that) and visually highlight them
- **notes or useful comments** these will be marked with `#NOTE` text to be recognized by editing platforms (most of IDEs platform have extensions for that) and visually highlight them

The words can appear anywhere is considered necessary: code, comments, README files, documentation, and so on.

For all previous enumerated markers and if they are used in code ghey will be prefixed by language specific comments code, for example in `Java`, `JavaScript` or `C` will be prefixed with `//` resulting for NOTE for example: `//#NOTE`.

Last update: July 18, 2023