# Differences in Project Requirements from NPR 7150.2C to NPR 7150.2D

This document provides the difference in project requirements from NPR 7150.2C to NPR 7150.2D. Requirements are shown in the order in which they appear in NPR 7150.2D. Only requirements with changed text are shown. Requirements with no changes or changes only to punctuation are not shown.

|  |  |
| --- | --- |
| **SWE Number** | **Requirements Text** |
| SWE-013 | The project manager shall develop, maintain, and execute software plans, including security plans, that cover the entire software life cycle and, as a minimum, address the requirements of this directive with approved tailoring. |
| SWE-024 | The project manager shall track the actual results and performance of software activities against the software plans. a. Corrective actions are taken, recorded, and managed to closure. b. Changes to commitments (e.g., software plans) that have been agreed to by the affected groups and individuals are taken, recorded, and managed. |
| SWE-027 | The project manager shall satisfy the following conditions when a COTS, GOTS, MOTS, OSS, or reused software component is acquired or used: a. The requirements to be met by the software component are identified. b. The software component includes documentation to fulfill its intended purpose (e.g., usage instructions). c. Proprietary rights, usage rights, ownership, warranty, licensing rights, transfer rights, and conditions of use (e.g., required copyright, author, and applicable license notices within the software code, or a requirement to redistribute the licensed software only under the same license (e.g., GNU GPL, ver. 3, license)) have been addressed and coordinated with Center Intellectual Property Counsel. d. Future support for the software product is planned and adequate for project needs. e. The software component is verified and validated to the same level required to accept a similar developed software component for its intended use. f. The project has a plan to perform periodic assessments of vendor reported defects to ensure the defects do not impact the selected software components. |
| SWE-015 | To better estimate the cost of development, the project manager shall establish, document, and maintain: a. Two cost estimate models and associated cost parameters for all Class A and B software projects that have an estimated project cost of $2 million or more. b. One software cost estimate model and associated cost parameter(s) for all Class A and Class B software projects that have an estimated project cost of less than $2 million. c. One software cost estimate model and associated cost parameter(s) for all Class C and Class D software projects. d. One software cost estimate model and associated cost parameter(s) for all Class F software projects. |
| SWE-151 | The project manager’s software cost estimate(s) shall satisfy the following conditions: a. Covers the entire software life cycle. b. Is based on selected project attributes (e.g., programmatic assumptions/constraints, assessment of the size, functionality, complexity, criticality, reuse code, modified code, and risk of the software processes and products). c. Is based on the cost implications of the technology to be used and the required maturation of that technology. d. Incorporates risk and uncertainty, including end state risk and threat assessments for cybersecurity. e. Includes the cost of the required software assurance support. f. Includes other direct costs. |
| SWE-018 | The project manager shall regularly hold reviews of software schedule activities, status, performance metrics, and assessment/analysis results with the project stakeholders and track issues to resolution. |
| SWE-022 | The project manager shall plan and implement software assurance, software safety, and IV&V (if required) per NASA-STD-8739.8, Software Assurance and Software Safety Standard. |
| SWE-141 | For projects reaching Key Decision Point A, the program manager shall ensure that software IV&V is performed on the following categories of projects: a. Category 1 projects as defined in NPR 7120.5. b. Category 2 projects as defined in NPR 7120.5 that have Class A or Class B payload risk classification per NPR 8705.4. c. Projects selected explicitly by the NASA Chief, Safety and Mission Assurance to have software IV&V. |
| SWE-131 | If software IV&V is performed on a project, the project manager shall ensure an IPEP is developed, approved, maintained, and executed in accordance with the IV&V criteria defined in NASA-STD-8739.8. |
| SWE-134 | If a project has safety-critical software or mission-critical software, the project manager shall implement the following items in the software: a. The software is initialized, at first start and restarts, to a known safe state. b. The software safely transitions between all predefined known states. c. Termination performed by software functions is performed to a known safe state. d. Operator overrides of software functions require at least two independent actions by an operator. e. Software rejects commands received out of sequence when execution of those commands out of sequence can cause a hazard. f. The software detects inadvertent memory modification and recovers to a known safe state. g. The software performs integrity checks on inputs and outputs to/from the software system. h. The software performs prerequisite checks prior to the execution of safety-critical software commands. i. No single software event or action is allowed to initiate an identified hazard. j. The software responds to an off-nominal condition within the time needed to prevent a hazardous event. k. The software provides error handling. l. The software can place the system into a safe state. |
| SWE-206 | The project manager shall require the software developers and custom software suppliers to provide NASA with electronic access to the models, simulations, and associated data used as inputs for auto-generation of software. |
| SWE-219 | If a project has safety-critical software, the project manager shall ensure that there is 100 percent code test coverage using the Modified Condition/Decision Coverage (MC/DC) criterion for all identified safety-critical software components. |
| SWE-220 | If a project has safety-critical software, the project manager shall ensure all identified safety-critical software components have a cyclomatic complexity value of 15 or lower. Any exceedance shall be reviewed and waived with rationale by the project manager or technical approval authority. |
| SWE-147 | The project manager shall specify reusability requirements that apply to its software development activities to enable future reuse of the software, including the models, simulations, and associated data used as inputs for auto-generation of software, for U.S. Government purposes. |
| SWE-148 | The project manager shall evaluate software for potential reuse by other projects across NASA and contribute reuse candidates to the appropriate NASA internal sharing and reuse software system. However, if the project manager is not a civil servant, then a civil servant will pre-approve all such software contributions; all software contributions should include, at a minimum, the following information: a. Software Title. b. Software Description. c. The Civil Servant Software Technical POC for the software product. d. The language or languages used to develop the software. e. Any third party code contained therein and the record of the requisite license or permission received from the third party permitting the Government’s use and any required markings (e.g., required copyright, author, applicable license notices within the software code, and the source of each third-party software component (e.g., software URL & license URL)), if applicable. f. Release notes. |
| SWE-157 | The project manager shall implement protections for software systems with communications capabilities against unauthorized access per the requirements contained in the Space System Protection Standard, NASA-STD-1006. |
|  |  |
|  |  |
| SWE-210 | The project manager shall identify software requirements for the collection, reporting, and storage of data relating to the detection of adversarial actions. |
| SWE-061 | The project manager shall select, define, and adhere to software coding methods, standards, and criteria. |
| SWE-135 | The project manager shall use static analysis tools to analyze the code during the development and testing phases to, at a minimum, detect defects, software security, code coverage, and software complexity. |
| SWE-065 | The project manager shall establish and maintain: a. Software test plan(s). b. Software test procedure(s). c. Software test(s), including any code specifically written to perform test procedures. d. Software test report(s). |
| SWE-071 | The project manager shall update the software test and verification plan(s) and procedure(s) to be consistent with software requirements. |
| SWE-194 | The project manager shall complete, prior to delivery, verification that all software requirements identified for this delivery have been met or dispositioned, that all approved changes have been implemented and that all defects designated for resolution prior to delivery have been resolved. |
| SWE-087 | The project manager shall perform and report the results of software peer reviews or software inspections for: a. Software requirements. b. Software plans, including cybersecurity. c. Any design items that the project identified for software peer review or software inspections according to the software development plans. d. Software code as defined in the software and or project plans. e. Software test procedures. |
| SWE-094 | The project manager shall provide access to the software measurement data, measurement analyses, and software development status as requested to the sponsoring Mission Directorate, the NASA Chief Engineer, the Center TAs, HQ SMA, and other organizations as appropriate. |
| SWE-203 | The project manager shall implement mandatory assessments of reported non-conformances for all COTS, GOTS, MOTS, OSS, and/or reused software components. |
| SWE-204 | The project manager shall implement process assessments for all high severity software non-conformances (closed loop process). |