# Kennedy NASA Procedural Requirements

**Effective Date:** **January 25, 2023**

**Expiration Date:** **January 25, 2028**

**Responsible Office:** **Engineering**

# NASA KSC Software Engineering Requirements

**National Aeronautics and Space Administration**

**John F. Kennedy Space Center**

KDP-KSC-T-2120 Rev Basic

**RELEASED - Printed documents may be obsolete; validate prior to use.**

**Change Log**

| Date | Revision | Description |
|---|---|---|
| 9/5/12 | Basic | New document |
| 12/2/14 | A | Updated to reflect the current procedural requirements for software engineering at KSC and to better align with the CMMI®-DEV model. Revisions include:<br>• References to NPR 7150.2 requirements (SWEs) were added to clarify that this KNPR defines the Center implementation of the requirements marked for Center tailoring (P.1, P.2, 1.2) and KSWE designations were removed.<br>• The role of the Technical Authorities (TA) and the procedure used in the requirements waiver process (P.2, 3.3)<br>• Document references were updated (P.4)<br>• Figure 1-1 was replaced with diagram depicting the flow-down of software requirements (1.4)<br>• Software Assurance and Software Safety requirements compliance with KNPR 8750.1 were added (3.1)<br>• Clarification of the Capability Maturity Model Integration (CMMI) rating requirement (SWE-032) for the software classes was added; rating requirements for Class C software is at the discretion of the Software TA (3.2)<br>• Information was included to fill gaps identified during CMMI® assessment in the following areas:<br>o Lifecycle definitions (2.2)<br>o Process tailoring (3.3)<br>• Software Support Requirements paragraph was added to reference Software Peer Review and Engineering Control of Software Products processes (3.13)<br>• Acronyms and definitions were updated to reflect changes in document (Appendices)<br>• Administrative changes to bring document into compliance with 1400.1, NASA Directives and Charters Procedural Requirements. |
| 10/19/18 | A-1 | • Throughout, changed Engineering and Technology Directorate to Engineering Directorate.<br>• NPR 7150.2 has a new revision and the appendices were changed. Updated those as well as removed the links to the Appendices. Other items in NPR 7150 had moved so references removed.<br>• NPR 7120.5 has a new revision so changed the link.<br>• NASA-HDBK-2203 has a new revision so changed the link.<br>• KSC-DES-0123 - Corrected the document name.<br>• Corrected the link to KDP-P-3917. |

| | | |
|---|---|---|
| | | • In Appendix C, section C.8, the KSC-UG-2805 Work Center Safety and Health Guide was cancelled and replaced with KFESH. Changed the example to KFESH. |
| 12/16/2019 | B | • Added KDP-P-3933 and KDP-P-3938<br>• Removed Appendix C, Appendix D, and Appendix E<br>• Updated document hierarchy diagram in Fig. 1-1<br>• Eliminated lifecycle definitions and referenced KDP-P-3916<br>• Clarified Process Tailoring Table and Compliance Matrix defined in SDP/SMP<br>• Revised Fig. 2.1-1<br>• Added Chapter 4 sections for website and KPAL<br>• Added the NASA Software Safety Guidebook in section 3.1 for guidance on safety critical software requirements. |
| 1/25/2023 | C | • Updated formatting to standardize across all EPG documents and to adhere to the KNPR format.<br>• NASA-STD- 8719.13 and KNPR 8750.1 have been cancelled and removed from document.<br>• Updated Applicability and Overview sections to include open-source software, embedded software, reused software, legacy software, and heritage software.<br>• Updated Software Development/Management Plan (SDP/SMP) Template to the latest revision.<br>• Added a reference to KSC-PLN-5400 which establishes and documents the implementation of Technical Authority (TA) at Kennedy Space Center (KSC).<br>• Updated the Safety Critical definition in Appendix A. |

**TABLE OF CONTENTS**

**Preface**
P.1 Purpose
P.2 Applicability
P.3 Authority
P.4 Applicable Documents and Forms
P.5 Measurement/Verification
P.6 Cancellation

**CHAPTER 1. Introduction**
1.1 Overview
1.2 Organizational Capability and Improvement
1.3 Roles and Responsibilities
1.4 Hierarchy of KSC Software-Related Documents

**CHAPTER 2. Project and Software Life Cycles**
2.1 NASA Project Life Cycle
2.2 Software Life Cycle

**CHAPTER 3. Software Engineering Requirements**
3.1 Compliance with NASA Procedural Requirement 7150.2, NASA Software Engineering Requirements
3.2 Project Formulation Requirements
3.3 Software Project Planning Requirements
3.4 Software Project Management Requirements
3.5 Software Requirements Definition Requirements
3.6 Software Design Requirements
3.7 Software Implementation Requirements
3.8 Software Integration and Test Requirements
3.9 Software Validation Requirements
3.10 Software Delivery Requirements
3.11 Software Sustainment and Change Management Requirements
3.12 Software Acquisition Requirements
3.13 Software Support Requirements
3.14 Organizational Software Support Requirements

**CHAPTER 4. Software Engineering Processes and Process Resources**
4.1 KSC Engineering Process Group Website
4.2 KSC Process Asset Library

**Appendix A. Definitions**
**Appendix B. Abbreviations and Acronyms**

## Preface

### P.1 Purpose

This document defines the software development and acquisition standards and practices that are required for National Aeronautics and Space Administration (NASA) employees and contractors who acquire, develop, or manage software for John F. Kennedy Space Center (KSC). This Kennedy NASA Procedural Requirement (KNPR) defines requirements for compliance with NASA Procedural Requirements (NPR) 7150.2, NASA Software Engineering Requirements, and identifies Center-level requirements and procedures. Software Engineering (SWE) Requirements refer to requirements identified in NPR 7150.2 and will be referred to in this document as SWE-XXX, where XXX refers to the requirement number identified in NPR 7150.2. This KNPR aids in meeting project level requirements per SWE-139 as marked in NPR 7150.2, Appendix C: Requirements Mapping Matrix and the Center level requirements per SWE-140. In the case where this document and NPR 7150.2 are in conflict, NPR 7150.2 takes precedence.

### P.2 Applicability

a. The requirements in NPR 7150.2 apply to software lifecycle activities started after its initial date of issuance (September 27, 2004). This KNPR establishes the center set of software requirements applicable to KSC projects for software development, testing, maintenance, retirement, operations, management, acquisition, and assurance activities. This includes Commercial-off-the-shelf (COTS) software, Government-off-the-shelf (GOTS) software, Modified-off-the-shelf (MOTS) software, open-source software, embedded software, reused software, legacy software, and heritage software. All software developed and acquired by or for NASA at KSC shall comply with NPR 7150.2.

b. The requirements in this KNPR apply to software lifecycle activities started after this KNPR's initial date of issuance (September 05, 2012) for all KSC programs, projects, subprojects, and tasks that are developing and/or acquiring software. This KNPR will be reviewed periodically and updated as needed to both incorporate changes to the Center's philosophy for compliance and to maintain alignment with updates to NPR 7150.2. The Software Engineering Technical Authority for Projects at KSC will ensure compliance with this KNPR.

c. Each project documents their planned compliance to the NPR 7150.2 requirements in their compliance matrix. If a project or program seeks relief from requirements specified by NPR 7150.2 consistent with program or project objectives, acceptable risk and constraints, they shall document these variances in their compliance matrix. The project's compliance matrix and any requirements tailoring with associated justification must be approved by designated Technical Authorities. NPR 7150.2, Appendix C, defines, for each requirement, the associated Technical Authorities that must approve tailoring. Refer to section 2.2 of NPR 7150.2 for the principles related to tailoring requirements.

d. In this directive, all mandatory actions (i.e., requirements) contain the term "shall." The terms "may" or "can" denote discretionary privilege or permission. "Should" denotes a good practice and is recommended, but not required. "Will" denotes expected outcome and "are/is" denotes descriptive material.

e. All document citations herein are assumed to be the latest version unless otherwise noted.



**RELEASED - Printed documents may be obsolete; validate prior to use.**

### P.3 Authority

a. NASA Policy Directive 7120.4, NASA Engineering and Program/Project Management Policy
b. NPR 7150.2, NASA Software Engineering Requirements
c. Kennedy NASA Policy Directive 2820.2, NASA KSC Software Policy

### P.4 Applicable Documents and Forms

a. NPR 7120.5 NASA Space Flight Program and Project Management Requirements
b. NPR 7120.7, NASA Information Technology Program and Project Management Requirements
c. NPR 7120.8, NASA Research and Technology Program and Project Management Requirements
d. NPR 7123.1, NASA Systems Engineering Processes and Requirements
e. NASA-STD-8709.22, Safety and Mission Assurance Acronyms, Abbreviations, & Definitions
f. NASA-STD-8739.8, Software Assurance and Software Safety Standard
g. NASA-HDBK-2203, NASA Software Engineering and Assurance Handbook
h. NASA-GB-8719.13, NASA Software Safety Guidebook
i. C-0029, KSC Engineering Process Group (EPG) Charter
j. KDP-D-1033, Procurement Process Model
k. KDP-P-3916, Software Project Planning
l. KDP-P-3917, Software Process Selection
m. KDP-P-3918, Software Project Monitoring and Control
n. KDP-P-3919, Software Classification
o. KDP-P-3920, Software Acquisition
p. KDP-P-3921, Software Requirements Development
q. KDP-P-3922, Software Preliminary Design
r. KDP-P-3923, Software Final Design
s. KDP-P-3924, Software Implementation
t. KDP-P-3925, Software Integration
u. KDP-P-3926, Software Verification Testing
v. KDP-P-3927, Software Validation Testing
w. KDP-P-3928, Software Delivery
x. KDP-P-3929, Software Sustainment
y. KDP-P-3930, Software Requirements Change Management
z. KDP-P-3931, Software Peer Review
aa. KDP-P-3932, Engineering Control of Software Products
bb. KDP-P-3933, Software Decision Analysis and Resolution
cc. KDP-P-3938, Software Project Estimation
dd. KDP-T-7150.2D_EPG_SDP_SMP, Software Development/Management Plan (SDP/SMP) Template
ee. KSC-DES-0123, Software Engineering Technical Authority for Class A Through E Software Projects at Kennedy Space Center (KSC)
ff. KSC-PLN-2302, Software and Systems Engineering Process Implementation (SSEPI) Plan



**RELEASED - Printed documents may be obsolete; validate prior to use.**

### P.5 Measurement/Verification

None

### P.6 Cancellation

This revision cancels KNPR 7150.2, Rev. B, NASA KSC Software Engineering Requirements.

**Approved by:**

*/original digitally signed by/*

_____

Shawn M. Quinn
Director, Engineering

Distribution: TechDoc Library



**RELEASED - Printed documents may be obsolete; validate prior to use.**

# CHAPTER 1. Introduction

## 1.1 Overview

This KNPR establishes the Center set of software requirements applicable to KSC projects for software development, testing, maintenance, retirement, operations, management, acquisition, and assurance activities. This includes Commercial-off-the-shelf (COTS) software, Government-off-the-shelf (GOTS) software, Modified-off-the-shelf (MOTS) software, open-source software, embedded software, reused software, legacy software, and heritage software. This document provides KSC specific guidance for projects containing software and refers to processes and templates to assist with compliance to NPR 7150.2. This document is maintained by the KSC Engineering Process Group (EPG) as chartered in C-0029.

## 1.2 Organizational Capability and Improvement

The Center shall maintain and implement the KSC-PLN-2302, Software and Systems Engineering Process Implementation (SSEPI) Plan, to continually advance its in-house software engineering capability and monitor the software engineering capability of KSC's contractors in accordance with requirement Software Engineering (SWE)-003.

## 1.3 Roles and Responsibilities

Roles and responsibilities are defined in KSC-PLN-2302.

## 1.4 Hierarchy of KSC Software-Related Documents

Figure 1-1 shows the relationship among the KSC software-related documents relevant to the flow-down of software engineering requirements.



**RELEASED - Printed documents may be obsolete; validate prior to use.**

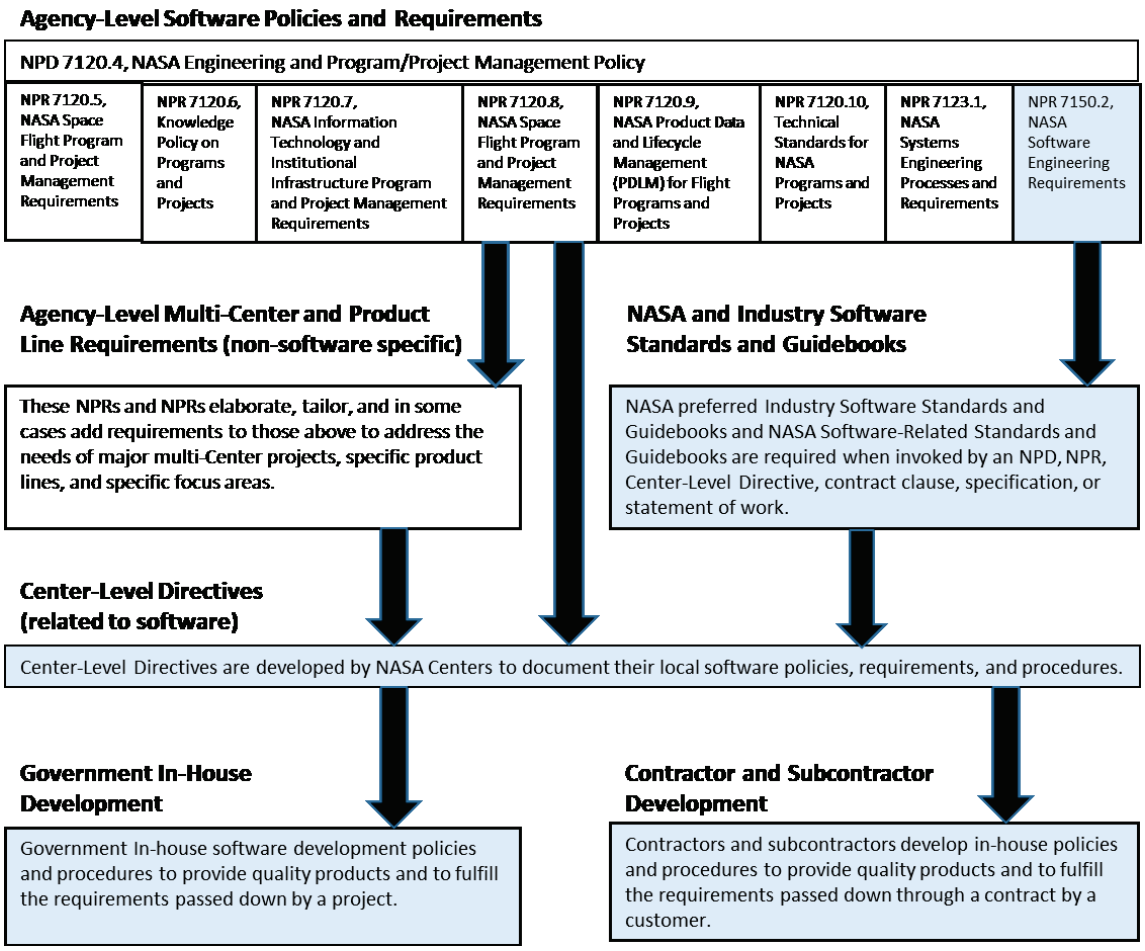



**Figure 1-1** Relationship of Governing Software Documents



**RELEASED - Printed documents may be obsolete; validate prior to use.**

# CHAPTER 2. Project and Software Life Cycles

## 2.1 NASA Project Life Cycle

The phases of the NASA project life cycle are shown in Figure 2.1-1 as defined in NPR 7120.5. The formulation and implementation periods are divided into seven phases. Each phase is typically marked by a key decision point, which usually is associated with a prescribed major design review. NPR 7120.5 provides additional detail about the NASA project life cycle, associated reviews, and key decision points and details for the formulation and implementation of space flight programs and projects. NPR 7120.7 provides details about the formulation and implementation of information technology and institutional infrastructure programs and projects. NPR 7120.8 provides details about the formulation and implementation of research and technology programs and projects.

| NASA Life-Cycle Phases | Approval for Formulation | FORMULATION | | Approval for Formulation | IMPLEMENTATION | | |
|---|---|---|---|---|---|---|---|
| **Project Life-Cycle Phases** | **Pre-Phase A:** Concept Studies | **Phase A:** Concept & Technology Development | **Phase B:** Preliminary Design & Technology Completion | **Phase C:** Final Design & Fabrication | **Phase D:** System Assembly, Integration & Test, Launch & Checkout | **Phase E:** Operations & Sustainment | **Phase F:** Closeout |

Figure 2.1-1 NASA Project Life Cycle

## 2.2 Software Life Cycle

The software life cycle is performed as part of the overall project or system life cycle, with an alignment of phases and milestones. Software reviews may be part of the system reviews or may be held separately, with the result of the software reviews incorporated into the system reviews. The phases of the KSC software life cycle and the relationship between Systems Engineering and Software Engineering processes are shown in Figure 2.2-1. The phases of the software life cycle are the same as those of the NASA program life cycle. Refer to KDP-P-3916, Software Project Planning, Attachment A for a discussion of NASA and Center approved lifecycles.

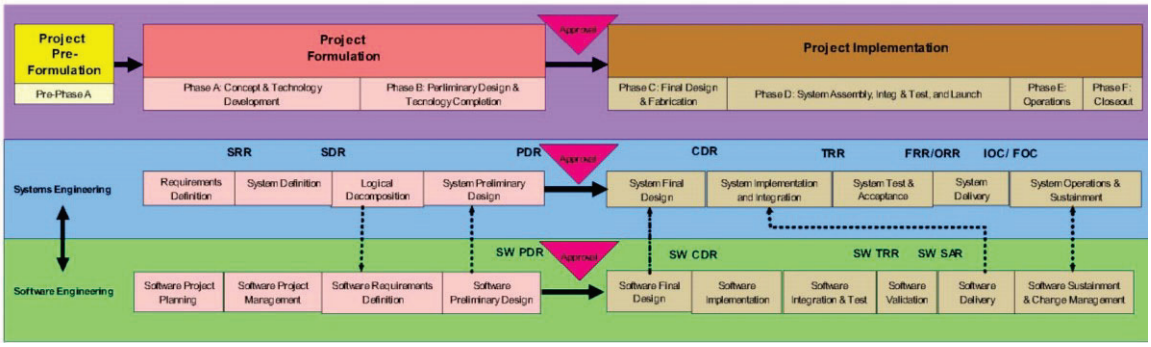



Figure 2.2-1 KSC Software Life Cycle



RELEASED - Printed documents may be obsolete; validate prior to use.

# CHAPTER 3. Software Engineering Requirements

## 3.1 Compliance with NASA Procedural Requirement 7150.2, NASA Software Engineering Requirements

3.1.1 The applicability of requirements in this KNPR to specific systems and subsystems containing software is determined through the use of the NASA-wide definitions for software classes and the designation of the software as safety-critical or non-safety-critical. Software classes are defined in Appendix D of NPR 7150.2. Refer to the NASA Software Engineering Handbook (NASA-HDBK-2203) for tools and guidance on applying these class definitions and assessing safety-criticality. It is available on the NASA Engineering Network (NEN).

3.1.2 A Software Assurance Classification Assessment of the software is performed to determine the safety criticality and classification of the software and the prioritization and level of software assurance needed. Projects shall classify their software into Classes in accordance with the software classification definitions of Appendix D in NPR 7150.2 and determine the software safety criticality in accordance with NASA-STD-8739.8, Software Assurance and Software Safety Standard.

3.1.3 The project shall ensure that the software assurance and software safety requirements in NASA-STD-8739.8 are implemented when the project is determined to have safety critical software and/or when software assurance is required.

3.1.4 Using the Requirements Mapping Matrix in NPR 7150.2, Appendix C, the project determines which SWE requirements are applicable to the project. Projects shall comply with all applicable requirements of NPR 7150.2 per SWE-139 as determined by their software classifications and safety criticality.

3.1.5 KSC-PLN-5400 establishes and documents the implementation of Technical Authority (TA) at Kennedy Space Center (KSC). The Center Director designates the Center Technical Authority(ies) for Projects who have approval authority for NPR requirement tailoring, according to the approval authority listed in NPR 7150.2, Appendix C. The roles and responsibilities for Technical Authorities are provided by NASA Headquarters' Office of the Chief Engineer and defined in NPR 7150.2. Delegation of KSC Software Engineering Technical Authority is authorized by the Center Director in accordance with directive KSC-DES-0123.

3.1.6 Requirements are expected to be tailored to accommodate the unique characteristics of the project. For tailored requirements, the project should discuss the chosen tailoring with the Software Engineering Technical Authority for Projects to obtain approval. The Software Engineering Technical Authority for Projects has final approval authority on the compliance for all software requirements delegated to the Center. As identified in the NPR, some requirements are assigned to an external Technical Authority (e.g., Headquarters Chief Engineer, Headquarters Office of Safety and Mission Assurance, etc.). When those requirements are tailored, approval must be obtained from the designated Technical Authority.

3.1.7 The project will use the Software Development/Management Plan (SDP/SMP) and associated NPR 7150.2 Compliance Matrix to document the requirement compliance. Refer to the template KDP-T-7150.2D_EPG_SDP_SMP, Software Development/Management Plan (SDP/SMP) Template, for guidance. When a project does not intend to fully implement a requirement, the rationale for this variance is documented in the project's compliance matrix. The compliance matrix and compliance



RELEASED - Printed documents may be obsolete; validate prior to use.

matrix updates shall be submitted to the Software Engineering Technical Authority for Projects.

3.1.8 The standard software process set has been developed as an engineering resource to implement NPR 7150.2 requirements for all classes of software at KSC. Projects that use the full standard set of software processes will be in compliance with NPR 7150.2. Projects may choose to build the project's process set using all of the standard processes or a subset of the standard processes, as long as the project's process set meets all NPR 7150.2 requirements applicable to the project's software classification and safety criticality. Projects may choose to tailor standard processes or replace them with alternate processes. Process tasks may be executed sequentially or in parallel, or process tasks may be repeated iteratively, depending on the design methodology being employed by the project. This process tailoring will be described in the SDP/SMP and documented in the SDP/SMP Appendix B, Defined Software Process Tailoring Table. Refer to the template, KDP-T-7150.2D_EPG_SDP_SMP. The Software Engineering Technical Authority for Projects at KSC must approve the Defined Software Process Tailoring Table. The project will work with the KSC EPG to determine the selection and tailoring of processes for the project's process set. The KSC EPG will provide recommendations to the Software Engineering Technical Authority for Projects at KSC for final approval of the process set selected by the project. Reference KDP-P-3917, Software Process Selection , for a description of process tailoring.

3.1.9 Each software process in the standard process set contains a mapping of each NPR 7150.2 SWE to the process block(s) of that process that implement(s) the SWE. SWEs that are not the responsibility of the project are not included in these matrices. This mapping can be used to determine which standard processes are needed to implement the project's applicable set of requirements. The mapping can also be used to identify the requirements that need to be addressed when a standard process is not used or is replaced by an alternate process. Refer to NPR 7150.2 for the definitions and full text of the SWEs (including applicable notes).

3.1.10 NASA-HDBK-2203, NASA Software Engineering Handbook, provides guidance for implementation of NPR 7150.2 requirements. It is available from the Software Engineering community on the NASA Engineering Network (NEN).

3.1.11 NASA-GB-8719.13, NASA Software Safety Guidebook, provides guidance for implementation of NPR 7150.2 requirements related to safety-critical software.

### 3.2 Project Formulation Requirements

3.2.1 The Capability Maturity Model Integration (CMMI) Institute has teamed with industry and government to develop the CMMI for Development model, which is a comprehensive integrated set of guidelines for applying best practices in a development organization. CMMI® rating requirements for software organizations that develop or maintain software for NASA are defined in NPR 7150.2 (SWE-032).

3.2.2 The project shall ensure that the organizations that are required to have a CMMI® rating maintain their software process capability as well as their CMMI® rating during the software development and maintenance periods, in accordance with SWE-032.

3.2.3 The standard process KDP-P-3919, Software Classification, contains the process for determining the software class and safety criticality of the software. This determination is used to assist the project in identifying applicable NPR 7150.2 requirements.



**RELEASED - Printed documents may be obsolete; validate prior to use.**

### 3.3 Software Project Planning Requirements

3.3.1 The project shall comply with NPR 7150.2 software project planning requirements applicable to the project's classification and safety criticality.

3.3.2 The following standard processes are resources that should be used to assist the project in complying with these NPR 7150.2 requirements:

a. KDP-P-3916, Software Project Planning
b. KDP-P-3917, Software Process Selection
c. KDP-P-3919, Software Classification
d. KDP-P-3933, Software Decision Analysis and Resolution
e. KDP-P-3938, Software Project Estimation

### 3.4 Software Project Management Requirements

3.4.1 The project shall comply with NPR 7150.2 software project management requirements applicable to the project's classification and safety criticality.

3.4.2 The standard process KDP-P-3918, Software Project Monitoring and Control, is a resource that should be used to assist the project in complying with these NPR 7150.2 requirements.

### 3.5 Software Requirements Definition Requirements

3.5.1 The project shall comply with NPR 7150.2 software requirements definition requirements applicable to the project's classification and safety criticality.

3.5.2 The following standard processes are resources that should be used to assist the project in complying with these NPR 7150.2 requirements:

a. KDP-P-3921, Software Requirements Development
b. KDP-P-3930, Software Requirements Change Management

### 3.6 Software Design Requirements

3.6.1 The project shall comply with NPR 7150.2 software design requirements applicable to the project's classification and safety criticality.

3.6.2 The following standard processes are resources that should be used to assist the project in complying with these NPR 7150.2 requirements.

a. KDP-P-3922, Software Preliminary Design
b. KDP-P-3923, Software Final Design

### 3.7 Software Implementation Requirements

3.7.1 The project shall comply with NPR 7150.2 software implementation requirements applicable to the project's classification and safety criticality.

3.7.2 The standard process KDP-P-3924, Software Implementation, is a resource that should be used to assist the project in complying with these NPR 7150.2 requirements.



**RELEASED - Printed documents may be obsolete; validate prior to use.**

### 3.8 Software Integration and Test Requirements

3.8.1 The project shall comply with NPR 7150.2 software integration and test requirements applicable to the project's classification and safety criticality.

3.8.2 The following standard processes are resources that should be used to assist the project in complying with these NPR 7150.2 requirements.

a. KDP-P-3925, Software Integration
b. KDP-P-3926, Software Verification Testing

### 3.9 Software Validation Requirements

3.9.1 The project shall comply with NPR 7150.2 software validation requirements applicable to the project's classification and safety criticality.

3.9.2 The standard process KDP-P-3927, Software Validation Testing, is a resource that should be used to assist the project in complying with these NPR 7150.2 requirements.

### 3.10 Software Delivery Requirements

3.10.1 The project shall comply with NPR 7150.2 software delivery requirements applicable to the project's classification and safety criticality.

3.10.2 The standard process KDP-P-3928, Software Delivery, is a resource that should be used to assist the project in complying with these NPR 7150.2 requirements.

### 3.11 Software Sustainment and Change Management Requirements

3.11.1 The project shall comply with NPR 7150.2 software sustainment and change management requirements applicable to the project's classification and safety criticality.

3.11.2 The following standard processes are resources that should be used to assist the project in complying with these NPR 7150.2 requirements.

a. KDP-P-3929, Software Sustainment
b. KDP-P-3930, Software Requirements Change Management

### 3.12 Software Acquisition Requirements

3.12.1 The project shall comply with NPR 7150.2 software acquisition requirements applicable to the project's classification and safety criticality.

3.12.2 The following standard processes are resources that should be used to assist the project in complying with these NPR 7150.2 requirements.

a. KDP-P-3920, Software Acquisition
b. KDP-D-1033, Procurement Process Model

### 3.13 Software Support Requirements

3.13.1 The project shall comply with NPR 7150.2 software requirements that support software activities and are applicable to the project's classification and safety criticality.



**RELEASED - Printed documents may be obsolete; validate prior to use.**

3.13.2 The following standard processes are resources that should be used to assist the project in complying with these NPR 7150.2 requirements.

a. KDP-P-3931, Software Peer Review
b. KDP-P-3932, Engineering Control of Software Products

### 3.14 Organizational Software Support Requirements

3.14.1 The organization shall comply with NPR 7150.2 software requirements that support development of Center wide processes and process resources.

3.14.2 The following standard processes are resources that should be used to assist the organization in complying with these NPR 7150.2 requirements.

a. KDP-P-7150.2_EPG_OPD, Software Process Definition
b. KDP-P-7150.2_EPG_OPF, Software Process Improvement
c. KDP-P-7150.2_EPG_OT, Software Organizational Training



**RELEASED - Printed documents may be obsolete; validate prior to use.**

# CHAPTER 4. Software Engineering Processes and Process Resources

## 4.1 KSC Engineering Process Group Website

The KSC EPG has developed the KSC EPG Website (https://epg.ksc.nasa.gov/) to provide access to the processes and process resources for software projects. The website provides links to the processes, templates, training, and process asset libraries that are of common value across projects at the Center and other information for process/product improvements developed and/or managed by the EPG. The website also provides a feedback mechanism for users to request help from the EPG, to provide process improvement suggestions, and to submit lessons learned and examples of best practices to the EPG.

## 4.2 KSC Process Asset Library

The KSC Process Asset Library (KPAL) is used to provide links to documents and data that reside in the EPG SharePoint and other repositories (e.g., Business World and KSC Design Data Management System) that are to be shared Center-wide. Assets to be included in the KPAL will be those determined by the KSC EPG to be of common value across projects at the Center, examples of best practices, or other information for process/product improvements. This is a “virtual” repository; the KSC EPG will maintain the KPAL Catalog file, which contains hyperlinks to the actual documents.



**RELEASED - Printed documents may be obsolete; validate prior to use.**

## APPENDIX A. DEFINITIONS

Acquisition A contract for developing software or software components that will be incorporated into a deliverable product or system.

Commercial Off-The-Shelf (COTS) Software Operating systems, libraries, applications, and other software purchased from a commercial vendor. Not customized for a particular project. Access to source code and documentation are often limited. (Definition from source document: NASA-STD-8709.22, Safety & Mission Assurance Acronyms, Abbreviations, & Definitions.)

Configuration Management A discipline applying technical and administrative direction, control, and surveillance to identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements.

Contractor An individual, partnership, company, corporation, association, or other service having a contract with the Agency for the design, development, manufacture, maintenance, modification, operation, or supply of items or services under the terms of a contract to a program or project within the scope of this KNPR. Research grantees, research contractors, and research subcontractors are excluded from this definition. (Definition from source document: NPR 7123.1, NASA Systems Engineering Processes and Requirements.)

Customer The organization or individual that has requested a product and will receive the product to be delivered. The customer may be an end user of the product, the acquiring agent for the end user, or the requester of the work products from a technical effort. Each product within the system hierarchy has a customer. (Definition from source document: NPR 7123.1.)

Deliverable Software Software developed and produced either as a contract end item or included as part of a deliverable end item (firmware/embedded software). It includes commercially available, reusable, or Government-furnished software, whether modified or unmodified, designated as part of a deliverable end item. (Definition from source document: NASA-STD 8709.22.)

Development The process of translating a design into software components (code). The creation and debugging of a software product from design documentation.

Firmware The combination of a hardware device and computer instructions and/or computer data that reside as read-only software on the hardware device. (Definition from source document: NASA-STD 8709.22.)

Government Off-the-Shelf (GOTS) Software Government-created software, usually from another project. The software was not created by the current developers (see software reuse). Usually, source code is included and documentation, including test and analysis results, is available, i.e., the Government is responsible for the GOTS software to be incorporated into another system. (Definition from source document: NASA-GB-8719.13.

Independent Verification & Validation (IV&V) Verification n and validation performed by an organization that is technically, managerially, and financially independent of the



RELEASED - Printed documents may be obsolete; validate prior to use.

development organization. (ISO/IEC 24765:2008 systems and software engineering vocabulary)

Legacy/Heritage Software products (architecture, code, requirements) written specifically for one project and then later found to be useful on other projects. See software reuse. (Definition from source document NPR 7150.2.)

Life Cycle The totality of a program or project extending from formulation through implementation, encompassing the elements of design, development, verification, production, operation, maintenance, support, and disposal. (Definition from source document: NASA-STD 8709.22.)

Modified Off-The-Shelf (MOTS) Software Product made from COTS, legacy/heritage software that is reused, or heritage software that is changed. The changes can include all or part of the software products and may involve additions, deletions, and specific alterations. An argument can be made that any alterations to the code and/or design of an off-the-shelf software component constitutes “modification,” but the common usage allows for some percentage of change before the off-the-shelf software is declared to be MOTS software. This may include the changes to the application shell and/or glueware to add or protect against certain features and not to the off-the-shelf software system code directly. See off-the-shelf software. (Definition from source document NPR 7150.2.)

Off-the-Shelf (OTS) Software Software not developed in-house or by a contractor for the specific project now underway. The software is generally developed for a different purpose than the current project. (Definition from source document NPR 7150.2

Open Source Software Software usually developed outside the project team, sometimes by an indefinite community of independent collaborators, for which the source code may be available to the general public. Users are (generally) able to view the source code, alter, and redistribute open source software.

Safety-Critical Any condition, event, operation, process, equipment, or system that could cause or lead to severe injury, major damage, or mission failure if performed or built improperly, or allowed to remain uncorrected. (Definition from source document NASA-STD-8739.8, Software Assurance and Software Safety Standard.)

Safety-Critical Software Software is safety-critical if it meets at least one of the following criteria:

- Resides in a safety-critical system (as determined by a hazard analysis) AND at least one of the following:
    - Causes or contributes to a hazard.
    - Provides safety-critical functions.
    - Controls safety-critical functions.
    - Processes safety-critical command or data.
    - Detects and reports, or take corrective action, if the system reaches a specific hazardous state.
    - Mitigates damage if a hazard occurs.
    - Resides on the same system (processor) as safety-critical software.



**RELEASED - Printed documents may be obsolete; validate prior to use.**

- Processes data or analyses trends that lead directly to safety decisions (e.g., determining when to turn power off to a wind tunnel to prevent system destruction).
- Provides full or partial verification or validation of safety-critical systems, including hardware or software subsystems.

(Definition from source document NASA-STD-8739.8

Software (SW) Computer programs, procedures, scripts, rules, and associated documentation and data pertaining to the development and operation of a computer system. Software includes programs and data. This also includes COTS, GOTS, MOTS, reused software, auto generated code, embedded software, firmware, and open source software components. (Definition from source document NPR 7150.2

Software Assurance The planned and systematic set of activities that ensure that software life cycle processes and products conform to requirements, standards, and procedures. For NASA, this includes the disciplines of Software Quality (functions of Software Quality Engineering, Software Quality Assurance, and Software Quality Control), Software Safety, Software Reliability, Software Verification and Validation, and IV&V. (Definition from source document: NASA-STD-8709.22.)

Software Classification A designation given to software based on project characteristics as defined in NPR 7150.2, Appendix D. The classification defines the requirements the project must meet during development of the software.

Software Products The documentation (plans, requirements, design, procedures, reports, etc.), code, executables, analysis results, intermediate analyses, and research, tests, models, simulators, prototypes, etc., created during the software development process. Many of these are for internal use in communicating what to create and how to go about it depending on the software classification.

Software Reuse A software product developed for one use but having other uses or one developed specifically to be usable on multiple projects or in multiple roles on one project. Examples include, but are not limited to, COTS products, acquirer-furnished software products, software products in reuse libraries, and existing developer software products. Each use may include all or part of the software product and may involve its modification. This term can be applied to any software product (such as requirements and architectures), not just to software code itself. Often this is software previously written by an in-house development team and used on a different project. GOTS software would come under this category if the product were supplied from one Government project to another Government project. (Definition from source document NASA-GB-8719.13.)

Validation Confirmation that the product, as provided (or as will be provided), fulfills its intended use. In other words, validation ensures that “you built the right thing.” (Definition from source document NPR 7150.2.)

Verification Confirmation that work products properly reflect the requirements specified for them. In other words, verification ensures that “you built it right.” (Definition from source document NPR 7150.2.)



**RELEASED - Printed documents may be obsolete; validate prior to use.**

## APPENDIX B. ABBREVIATIONS AND ACRONYMS

| | |
|---|---|
| CDR | Critical Design Review |
| CMMI® | Capability Maturity Model Integration |
| COTS | Commercial Off-The-Shelf |
| EPG | Engineering Process Group |
| FOC | Full Operational Capability |
| FRR | Flight Readiness Review |
| GOTS | Government Off-The-Shelf |
| IOC | Initial Operational Capability |
| IV&V | Independent Verification and Validation |
| KDDMS | KSC Design Data Management System |
| KDP | Kennedy Documented Procedure |
| KNPR | Kennedy NASA Procedural Requirements |
| KSC | Kennedy Space Center |
| MOTS | Modified Off-The-Shelf |
| NASA | National Aeronautics and Space Administration |
| NEN | NASA Engineering Network |
| NPR | NASA Procedural Requirements |
| ORR | Operational Readiness Review |
| OSS | Open Source Software |
| OTS | Off-The-Shelf |
| PDR | Preliminary Design Review |
| SAR | System Acceptance Review |
| SDP/SMP | Software Development/Management Plan |
| SDR | System Definition Review |
| SRR | System Requirements Review |
| SSEPI | Software and Systems Engineering Process Implementation |
| SW | Software |
| SWE | Software Engineering |
| TRR | Test Readiness Review |



**RELEASED - Printed documents may be obsolete; validate prior to use.**