**Software Development Plan for the *[identify project]***

|  |
| --- |
| ***[identify the division being support]***OPEN TO JSC/JSC CONTRACTOR EMPLOYEES & OTHER NASA/NASA CONTRACTOR EMPLOYEES, AS REQUIRED**[Mm/dd/yyyy]** |

|  |
| --- |
| **APPROVAL SHEET** |
|  |
| **Software Development Plan for the *[ Identify the project ]*** PREPARED BY: Name DATE Title APPROVED: Name DATE Title |

|  |  |  |
| --- | --- | --- |
| **DOCUMENT NUMBER**xxxxxxx | **DOCUMENT CHANGE/****REVISION LOG** | **PAGE** 1 **OF** 1 |
| **CHANGE/REVISION** | **DATE** | **DESCRIPTION OF CHANGE** | **PAGESAFFECTED** |
| − | Xx/xx/xx  |  Baseline | ALL |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

CONTENTS

Section Page

1.0 INTRODUCTION 1-1

1.1 SCOPE 1-1

1.2 RESPONSIBILITY AND CHANGE AUTHORITY 1-1

2.0 RELATED DOCUMENTATION 2-1

2.1 APPLICABLE DOCUMENTS 2-1

2.2 REFERENCED DOCUMENTS 2-1

2.3 ORDER OF PRECEDENCE 2-2

3.0 ROLES AND RESPONSIBILITIES 3-1

3.1 SOFTWARE TEAM 3-1

3.2 ADDITIONAL RELEVANT STAKEHOLDERS 3-2

4.0 PROJECT-SPECIFIC SOFTWARE TRAINING 4-1

5.0 SOFTWARE DEVELOPMENT METHODOLOGY AND APPROACH 5-1

5.1 SOFTWARE FUNCTIONAL OVERVIEW 5-1

5.2 SOFTWARE LIFECYCLE DEVELOPMENT METHODOLOGY AND APPROACH 5-1

5.2.1 Software Classification 5-1

5.2.2 Software Versioning 5-1

5.2.3 Planning for Lifecycle Exceptions 5-2

5.3 SOFTWARE ACQUISITION STRATEGY 5-2

5.4 SOFTWARE DEVELOPMENT SCHEDULE 5-3

5.4.1 Project Phases 5-3

5.4.2 Documentation Tree 5-5

5.4.3 Reviews 5-6

5.4.4 Deliverables 5-6

5.5 SOFTWARE STANDARDS 5-7

5.6 SOFTWARE DEVELOPMENT ENVIRONMENT (TOOLS, FACILITY, GROUND) 5-7

5.7 PROCESS AND PRODUCT INSPECTIONS, PEER REVIEWS, AND WALK THROUGHS 5-7

6.0 SOFTWARE CONFIGURATION MANAGEMENT PLAN 6-1

6.1 CONFIGURATION IDENTIFICATION 6-1

6.1.1 Documentation Identification 6-2

6.1.2 Software Identification 6-2

6.1.3 Development Environment and Tool Identification 6-2

6.1.4 Electronic File Identification 6-2

6.2 CONFIGURATION CONTROL AND ACCOUNTING 6-2

6.2.1 Configuration Control 6-2

6.2.2 Control, Release, and Accounting 6-3

6.2.3 Control, Release, and Accounting of Electronic Files 6-4

6.3 CONFIGURATION VERIFICATION 6-4

7.0 DATA MANAGEMENT PLAN 7-1

8.0 SOFTWARE ASSURANCE PLAN 8-1

8.1 SOFTWARE QUALITY 8-1

8.2 AUDITS OF SOFTWARE PROCESSES AND PRODUCTS 8-1

8.2.1 Audit Process 8-1

8.2.2 List of Audits to be Performed 8-2

8.3 SOFTWARE VERIFICATION AND VALIDATION 8-3

9.0 SOFTWARE MAINTENANCE/SUSTAINING ENGINEERING PLAN 9-1

9.1 MAINTENANCE PROCESS IMPLEMENTATION 9-1

10.0 SOFTWARE RETIREMENT PLAN 10-1

Appendix A 10-1

LIST OF TABLES

Table Page

[Table 5‑1. Software Documentation Deliverables 5-6](#_Toc471992172)

[Table 7‑1. Data Management Plan 7-1](#_Toc471992173)

[Table 9‑1. Process Audits to be Performed 9-2](#_Toc471992174)

[Table 9‑2. Product Audits to be Performed 9-2](#_Toc471992175)

LIST OF FIGURES

Figure Page

[Figure 5‑1. Documentation Tree 5-5](#_Toc471992181)

*This annotated outline is intended to assist the Software Lead/Developer in developing the Software Development Plan (SDP) for a flight software project, implementing the policies and practices required for this phase of the flight software development process.*

*The wording in italics is intended for information only and should not appear in the final document. The wording not in italics should be used in the final document verbatim. This font convention is applicable throughout the document (including the front and signature pages). Note that some sections contain only explanatory text for the document author and may not necessarily contain any text in the final document.*

# INTRODUCTION

*Describe the purpose, how the data for the project will be captured and provide a general description for the flight software project. An example is provided below.*

This document describes the ***[system…….. ]*** for use by the Human Research Program (HRP), which is being developed by ***[organization …]***

## SCOPE

This Software Development Plan (SDP) documents the development plan for the ***[system…….. ]*** As used throughout this plan, the term software refers to both software and firmware. This SDP includes plans for software configuration management, software assurance, software maintenance, and software retirement.

## RESPONSIBILITY AND CHANGE AUTHORITY

*The planning for the software is being docucmented via this Sofware Development Plan (SDP). One of the first steps involved in defining all the requirements and products necessary according to NASA is to classify the software. The software is classified according to the type of function and usage of the software. Examples and the different types of classifications are documented in JPR 7150.2 , Appendix C. For software that is flown or used on the ISS* *is either Class A, B or C. Depending on the usuage, usually for a research project, it is classified as Class C.*

This document is prepared in accordance with JSC Procedural Requirement (JPR) 7150.2, Johnson Space Center Software Engineering Procedural Requirements (JSWE-13), and tailored for a Class C Software in support of the Health and Human Performance Directorate (HHPD). The responsibility for the development of this document lies with the ***[identify the organization]****.*

#

# RELATED DOCUMENTATION

The following documents are referenced in this plan. Unless otherwise specified, the exact issue shown is the applicable version.

## APPLICABLE DOCUMENTS

The following documents, of the exact issue and revision shown, form a part of this SDP to the extent specified herein.

|  |  |  |
| --- | --- | --- |
| **Document Number** | **Revision/ Release Date** | **Document Title** |
| JPR 7150.2 | Rev. A01/23/2017 | Johnson Space Center (JSC) Software Engineering Procedural Requirements |
|   | Revision x mm/dd/yy |   |
|  |  |  |
|  |  |  |
|  |  |  |

## REFERENCED DOCUMENTS

The following documents are referenced within this plan**. [add applicable documents ..]**

|  |  |  |
| --- | --- | --- |
| **Document Number** | **Revision/ Release Date** | **Document Title** |
| SSP 50313 | Revision xmm/dd/yy | Display and Graphics Commonality Standard |
|   | Revision x mm/dd/yy |   |
|  |  |  |
|  |  |  |
|  |  |  |

## ORDER OF PRECEDENCE

An exception to JPR 7150.2 requires a waiver approved by the JSC Software Engineering Process Group and JSC Software Reliability and Quality Assurance technical authority (JSC/NA).

# ROLES AND RESPONSIBILITIES

## SOFTWARE TEAM

*The common software development roles are described below. Describe your (PI) software development roles and their responsibilities for the project team in this section.*

The **Principal Investigator (PI)** is the individual or team that obtained NASA funding to perform human subject research on ISS

The **Software Development Team** performs the software development and integration activities defined, under the direction of the Technical Lead. They develop the software architecture and design. They implement the design into software and firmware and assist in the development of the verification and validation (V&V) procedures.

The **Software Test Team** performs the software verification and validation activities, under the direction of the Technical Lead. They develop the software testing approach and method of test needed to verify the requirements. They work with the Software Development Team in the development of the test plan and procedures and conduct the tests.

The **Software Configuration Management Administrator** is responsible for maintaining configuration control of the software products. This includes repository creation, and access controls for each repository based on each users role.

The **Software Assurance (SA) Representative** performs software quality assurance, software quality engineering, and software quality control. They perform assessments to assure process and product adherence and evaluate and monitor work products [documents, test plans, procedures and reports), and discrepancy reports, change records and corrective actions.

## ADDITIONAL ISSMP RELEVANT STAKEHOLDERS

*Describe the planned involvement of additional relevant ISSMP stakeholders including Crew, Mission Operations, Mission Training, Increment Leads, Experiment Teams, Hardware Teams, Payload Support and specialized facility personnel, if they are known as this time.. Below are common relevant stakeholders. Modify accordingly.*

* Experiment Teams
	+ Flight Project Manager (FPM) –is the lead for the Experiment Support Team (EST) and is responsible for interfacing with the NASA Technical Monitor, the Principal Investigator (PI), and other external groups necessary to implement the experiment.  The FPM is responsible for managing the EST and has ultimate responsibility over successful experiment development, integration, and operations.  The FPM is responsible for the coordination of experiment resource requirements, experiment reviews, software integration, and risks/opportunities.  The FPM coordinates experiment resource requirements with ISSMP resource availability and supports conflict resolution.
	+ Experiment Support Scientist (ESS) –serves as the primary scientific liaison between the PI team and various NASA organizations throughout the entire experiment life cycle.  The ESS will provide inputs for software development, support meetings, develop experiment specific Payload Display Review Team / Integrated Display and Graphics Standards (PDRT/IDAGS) documentation, coordinate Baseline Data Collection (BDC), plan and conduct crew training and on-orbit operations, and perform close-out activities at experiment conclusion.
* Hardware Teams
	+ Hardware Support Engineer (HSE) –serves as the primary liaison between the Hardware Team, Software Integration Team and experiment teams.  The HSE will provide inputs for software development, support meetings, and support on-orbit operations.
* The ISSMP **Software Development Team** performs the flight software integration activities to ensure the software is integrated into the ISS system using a flight like simulated test environment as expected. They assist in the development of the verification test procedures and conduct the verification test activities for certification. They also assist with the generation of the certification products necessary for the flight software

.

# PROJECT-SPECIFIC SOFTWARE TRAINING

No specific training is required for the project. (JSWE-17)

# SOFTWARE DEVELOPMENT PLAN

## SOFTWARE FUNCTIONAL OVERVIEW

*Describe the purpose and major functions of the software being developed. In this section, include only the level of detail needed to establish the scope and context of the software development activities.*

The **[ project …]** software comprises the following functions:

*Below is an example to use as a guide:*

*The HRF (Human Research Facility) software comprises the following software used to:*

* *Extend rack interfaces to experiments and sub-rack devices*
* *Support a specific rack device*

*Examples include but are not limited to:*

* *HRF Common Software (CSW) – Custom software installed on each HRF laptop and provides the following capabilities:*
	+ *Rack/Payload Data Interface*
	+ *Crew/User Interface (UI)*
	+ *File Management*

## SOFTWARE LIFECYCLE DEVELOPMENT METHODOLOGY AND APPROACH

*Summarize the process for developing the software. State any known constraints that affect the software development methodology and approach. Describe the selected development methodology, such as object-oriented, procedural, incremental delivery, or the use of Unified Modeling Language (UML).*

 The **[ project…]** will use the ***{waterfall, iterative, agile, etc.}*** life cycle process model for the development of the software. (JSWE-19)

## Software Classification

The software has been assessed as Class C, non-safety critical software, in accordance with JPR 7150.2. The classification and safety determination is documented on the JSC Form (JF) 1704. (JSWE-20, 21, 132, and 133) The approved JF 1704 forms and accompanying compliance matrices are located in Appendix A and B of this document.

Any change to the scope or functionality of the software will require a re-assessment of the software classification and safety criticality per JPR 7150.2.

## COMPLIANCE TO JPR 7150.2A

*The process for completing this mapping includes evaluating each requirement for applicability to the software based on the classification and documenting those results of the evaluation in the JPR 7150.2 Compliance Matrix.*

In accordance with JPR 7150.2, a compliance matrix is completed to ensure compliance against the requirements identified for that software classification. (JSWE-125). This matrix is located in Appendix B.

## Planning for Lifecycle Exceptions

*Projects often don’t go according to the original plan. As changes to this plan become necessary, ensure that that changes are agreed to by the affected groups and individuals and relevant stakeholders. Identify the process or method.*

As changes to this plan become necessary, a **[identify the process ….]** to ensure that changes are agreed to by the affected groups, individuals and stakeholders.

## SOFTWARE ACQUISITION STRATEGY

*Software development doesn't always mean new development. Sometimes a commercially available software product can be used or exists that can be modified to meet the project's requirements. In any case, the options for acquisition must be assessed against a set of criteria including design applicability, risk, and cost and compared to determine the benefits of one option against another.*

*When considering COTS, MOTS, open source, reuse, legacy, or heritage software (collectively known as OTS software), the following conditions should be considered:*

*a) The requirements that are to be met by the OTS software.*

*b) The OTS software includes documentation to fulfill its intended purpose (e.g., usage instructions).*

*c) Proprietary, usage, ownership, warranty, licensing rights, and transfer issues of the OTS software.*

*d) Future support for the OTS software product*.

*e) The OTS software is validated to the same level of confidence as would be required of the developed software.*

*If procuring OTS software, validating the software should be the same level of confidence that would be needed for an equivalent class of software if obtained through a "development" process.*

*Supplier agreement to deliver the source code or third party maintenance agreement is in place.*

*A risk mitigation plan to cover the following cases is considered:
1) Loss of supplier or third party support for the product.
2) Loss of maintenance for the product (or product version).
3) Loss of the product (e.g., license revoked, recall of product, etc.)*

*Agreement that the project has access to defects discovered by the community of users has been obtained. When available, the project can join a product users group to obtain this information.*

*The plan to provide adequate support is in place, including timely maintenance and cost of maintenance.*

*Any changes to the software development, or maintenance plans that are affected by the use or incorporation of COTS, GOTS, MOTS, reuse, legacy, or heritage software should be documented by the project.*

*Identify the method or strategy in this section, otherwise, state that no software will be acquired for the project.*

## SOFTWARE SIZE

*Estimate the number of SLOCs (source lines of code) or estimated size of the effort (small, medium or large) for each of the software. Use the following table as a guide.*

|  |  |
| --- | --- |
| Developed, internally | *x* |
| Developed, contracted | *x* |
| Acquire Off-the-shelf (OTS) | *x* |
| Open source | *x* |
| Reuse, legacy, or heritage | *x* |
| Acquire and Modify Off the shelf software | *x* |
| Total Software Size | *Put total here* |

## Software Development Schedule

*The software development schedule will document milestone reviews, precedence’s, critical paths, deliverables and describe the overall project schedule. Use the following table below as a guide. Keep in mind that this will be a preliminary schedule to be used as a starting point. The schedule will need to be reviewed by ISSMP and which may be updated and revised as necessary to integrate with program, flight, experiment, and other schedules for flight..*

The software-specific schedule will be within the project schedule (SWE-016). The software specific schedule will:

Identified below is the initial schedule for the planning of the software development at this time for baseline. The current scheduled will be maintained ***[Identify the location ]***.

| ***Task*** | ***Initial Planning Date*** | ***Milestone*** |
| --- | --- | --- |
| *Completion of the JF 1704* | *Xx/xx/xx* | *Product, baseline at Requirements Review in SDP* |
| *Baseline of the Software Development Plan (SDP)*  | *Xx/xx/xx* | *Product, Baseline at Requirements Review* |
| *JPR 7150.2A Compliance Matrix* | *Xx/xx/xx* | *Product, baselined at Requirements Review and updated as needed as software matures* |
| *Requirements Review (i.e. SRR)* | *Xx/xx/xx* | *Milestone review*  |
| *Release of the Software Requirement Specification (SRS)* | *Xx/xx/xx* | *Product, baselined after Requirements Review*  |
| *Design Review (i.e. PDR, CDR)* | *Xx/xx/xx* | *Milestone review*  |
| *Release of Software Test Plan (STP) and Software Test Procedures* | *Xx/xx/xx* | *Product, baseline after Design Review and prior to TRR* |
| *Release of Software Test Report (STR)* | *Xx/xx/xx* | *Product, before acceptance of the software* |
| *Acceptance Review*  | *Xx/xx/xx* | *Milestone for acceptance of software*  |
| *Version Description Document (VDD)* | *Xx/xx/xx* | *Product, baseline due at acceptance of software* |

## Project Documentation Tree

*Word document templates can be found on the PI Website in the template folder to aid the development of the deliverable documentation.*

The deliverable documentation for the software is as follows.

## Project Reviews

The reviews associated with the project are identified in the following sections.

### Requirements Review

*The purpose of the requirements reviews is to foster communication among all members involved in analyzing the performance of the application and to define the functionality of the software in the form of requirements to be met and objectives to achieve. Typically, this type of review will be in the form of a System or Software Requirements Review (SRR) is held. For smaller projects, it can also be in the form of a Technical Interface Meeting (TIM). The requirements and objectives will serve as a basis for estimating project costs and for acceptance of the software which are documented in a Software Requirements Specification (SRS). The test plan and procedures will trace back to those requirements identified in the Software Requirements Specification (SRS) in the form of a traceability matrix used in the verification of the requirements during Testing. Identify the milestone reviews that will be held or process used to identify and agree upon the requirements with the relevant stakeholder and affected personnel, including the Experiment Lead and ISSMP Integration Team.*

### Design Reviews

*The purpose of the design reviews is to ensure the software meets the objectives and goals for the project being developed. Prototypes of the software can be used to demonstrate the user interface. Typically a Preliminary Design Review (PDR) and a Critical Design Review (CDR) will be conducted for the project. Each design review will further refine the requirements and design to ensure the design will met the intended function, objective and needs.*

*Development starts following authority to proceed. The development phase includes creation or update of the software architecture, design, code and unit testing of the software. The software products are developed using the templates provided in the PI website in the template folder. Identify the milestone design reviews that will be held or process used to demonstrate or refine the design with the relevant stakeholder and affected personnel, including the Experiment Lead and ISSMP Integration Team.*

#### Testing

Testing ensures that the software is being developed to satisfy the functional, performance, and other system requirements and yields the right product.

The testing activities:

a) Assure that the project complies with their documented software test plan.

b) Form a part of acceptance

The Software Development Team along with the Software Test Team are responsible for verifying that all requirements are met, either through inspection, demonstration, analysis, and/or test.

*Identify the test approach that will be used, the required personnel (i.e. if user acceptance testing or user testing will occur, the phase that the test will occur (i.e., unit, system integration, user acceptance, release).*

### Reviews

Payload Display Review Tean (PDRT) and ISS Display and Graphics Standards (IDAGs) Review

*A screen shot of the crew user interface display or a demonstration of the software display will be submitted to two display review teams, PDRT and the IDAGs for review. First the payloads team will review the displays and then coordinate with the ISS team to review and provide comments regarding the displays. . Either approval of the displays will be given, or recommendation for change or update to the displays will be provided, or a waiver will be obtained for any deviations needed against the commonality standard for the crew/user interface. The review process will begin as early in the design process as possible to avoid design impacts late in the software development schedule.*

A IDAGs review is conducted in accordance with external processes to ensure the standard for the crew interface is met per SSP 50313, ISS Display and Graphics Commonality Standard, and any deviations are identified and approved.

## Deliverables

*The software source code deliverable items will be identified in the Version Description Document (VDD) along with the source code repository location. However, in order to deliver the source code deliverable items to the ISSMP Integration Team, the source of the file needs to be identified on the Certificate of Conformance (CofC). Below is an example of the software deliverable documentation to be used as a guide.*

The software product documentation that will be delivered are provided below.

| **Document** |  **Format** | **Location** |
| --- | --- | --- |
| Software Development Plan (SDP) | Template format available on the PI website | ***[identify the location]*** |
| Software Requirement Specification (SRS) | Template format available on the PI website | ***[identify the location]*** |
| Version Description Document (VDD) | Template format available on the PI website | ***[identify the location]*** |
| Software Test Plan (STP) | Template format available on the PI website | ***[identify the location]*** |
| Software Test Report (STR) |  | ***[identify the location]*** |
| Certificate of Compliance (CofC)  | Template format available on the PI website | N/A |
| JF 1704 | Form available on the PI website | N/A |
| JPR 7150.2 Compliance Matrix | Form available on the PI website | N/A |

## SOFTWARE STANDARDS

*Provide any coding, styling standards that are used in the development of the software code. (JSWE-061)*

## SOFTWARE DEVELOPMENT ENVIRONMENT (TOOLS, FACILITY, GROUND)

*Provide the environment, tools, supporting equipment that is necessary during the development of the software code. Use the table below as a guide.*

| **Facility** | **Environment** | **Use** |
| --- | --- | --- |
| Devleopment Lab |   | Development/Code Coverage |
| Testing Lab |   | Demonstration/Unit Test/Validation |
| Facility  |   | Unit Test/Demonstration/Verification/Validation/Calibration |

.

# SOFTWARE CONFIGURATION MANAGEMENT PLAN (JSWE-079)

Configuration Management (CM) is composed of the following basic functions:

1. configuration identification
2. configuration control and accounting

## CONFIGURATION IDENTIFICATION

Configuration item identification consists of the physical identification of project deliverables, including items such as documentation, software, firmware, environment and tool, and electronic files.

### Documentation Identification

All CM-controlled documents will be identified via a unique document number as assigned by the technical documentation organization.

### Software Identification

Software identification uniquely identifies and names all source code files, header files, and derived products (e.g., executable). For all software files, a unique name is used along with the applicable software configuration management tool version rules to uniquely identify them. For each derived-product file, a version-specific name is used that uniquely identifies it to the user and allows for ease in the configuration management of these items.

The software code version identification is document in the Version Description Document (VDD).

### Software Versioning

*Establish a software version naming and numbering convention.*

*The most common software versioning scheme is a scheme in which different releases of the software receive a unique numerical identifier. This identifier is typically expressed as three numbers, separated by periods. The format may be one of the following:*

*major.minor*

Each release of the software will have a unique version number using the format **[*x.y*. ].**

### Development Environment and Tool Identification

*The development environment and tool identification uniquely identifies all computers, peripheral hardware, hardware and software tools (e.g., compiler, linker, etc.), and make file/build scripts used in the development, build, test, and installation of the software. Each item has a version-specific name that uniquely identifies it to the user and allows for ease of CM of these items. In some cases, the name is used along with the applicable software CM tool version rules to uniquely identify the item*.

The configuration of the development environment for software development, build, test, and installation is uniquely identified for each version of the software delivered and documented in that version’s VDD.

### Electronic File Identification

A unique file name and date is used to identify each electronic file.

## CONFIGURATION CONTROL AND ACCOUNTING

Configuration control and accounting consists of tracking and managing all changes to documentation, software, firmware, environment and tool, and electronic files.

### Configuration Control

#### Configuration Control Authority

Configuration control and accounting consists of tracking and managing all changes to documentation, software, firmware, environment and tool, and electronic files.

#### Configuration Control Process

The configuration control process consists of two distinct functions. The first is called Version Control and consists of the tracking of different versions of a configured item. The second is called Change Control and consists of tracking the reason for a change along with implementation details.

*Identify the method or tool that will be used to version control the versions of the software. Describe the method or tool that will be used to control and track changes to the software after the baseline of the requirements have been defined or any modifications after the baseline release of the software.*

#### Problem Reporting

*Describe any bug reporting tool or process that will be used for the identification of issues or discrepancies during the development of the software.*

### Control, Release, and Accounting

All software deliverables and internal products will be maintained under some level of CM. The following sections describe the details pertaining to the different classes of products.

#### Control, Release, and Accounting of Documentation

The VDD will be generated, managed and processed in accordance to LS-73001, ISSMP Configuration Management Plan per the ISSMP standard processes using their provided template.

*Describe the method or process that will be used for the control and release of the software product documentation.*

#### Control, Release, and Accounting of Software Code

Configuration control, release, and accounting of project software, development environment, and tools through the project’s lifecycle*.* Software includes all source code files, header files, data files, and derived products (e.g., object code, executable, libraries, etc.).

After the software is placed under configuration control, the following will be performed:

* 1. The software release is placed in a software CM-controlled repository that will be backed up on a regular basis.
	2. The software development tools are maintained by the developer.
	3. A unique software release or version number is used to identify each version or revision of the software.

### Control, Release, and Accounting of Electronic Files

*Describe how or with what tool the release versions of software are controlled either with electronic files or controlled electronic media. Also describe who will have administrative privileges for the release of the files.*

A Certificate of Conformance (CofC) is delivered with the software when the software is delivered to ISSMP for acceptance and integration.

*(Refer to the Experiment Lead for information and a template for the CofC located on the PI website)*

# DATA MANAGEMENT PLAN

The data management plan defines how miscellaneous items (e.g., meeting minutes, action items, review documentation, etc.) will be controlled. Identified below are the data products that will be managed, who has control for those items and were they will be located.

*Update the table below according to the products that will be managed.*

| **Work Product** | **Storage Location** | **Level of Control** |
| --- | --- | --- |
| Action Items |   |   |
| Informal problem reports |   |   |
| Documentation to be Reviewed |   |   |
| Software/Project Plans and Schedules |   |   |
| Deliverable Documentation |   |   |

# SOFTWARE ASSURANCE PLAN (JSWE-022)

*Describe the functions and audits/assessments that the software quality assurance personnel will be performing during the development, acceptance and delivery of the software deliverables and software deliverable documentation. Below is an example.*

The software assurance functions that are performed by software assurance include software quality (quality assurance (process and product) and quality engineering), software verification and validation.

## SOFTWARE QUALITY

Software quality includes software quality assurance, quality control, and quality engineering. The software quality personnel perform the following functions.

1. Perform audits of process requirements (e.g., project planning, requirements management) at designated reviews.
2. Approve test procedures to ensure quality control processes are met.
3. Witness tests.
4. Ensure that test software and test data are ready for use.
5. Participate in reviews to ensure that software products meet requirements.
6. Review software non-conformances and problem reports
7. Review any changes to the baselined, as built, or updates to software product.
8. Review and approve all exceptions and deviations.

## AUDITS OF SOFTWARE PROCESSES AND PRODUCTS

### Audit Process

*Identify any process that will be conducted in performing audits that will be conducted by the software quality assurance personnel*. *Use the tables provided below as a guide.*

### Process Quality Audits

The table below identifies the processes that will be audited along with the corresponding phases.

|  |  |
| --- | --- |
| **Process** | **Project Lifecycle Phase** |
| **Requirements.** |  **Design** | **Testing** | **Delivery** | **Acceptance** |
| Requirements Management | - | √ | √ | √ | √ |
| Process and Product Quality Assurance | √ | √ | √ | √ | √ |

### Product Quality Audits

The table below identifies the product audit that will be performed by the software quality assurance personnel.

| **Product** | **Requirements Definition** | **Delivery** | **Acceptance** |
| --- | --- | --- | --- |
| Software Development Plan | √ | - | - |
| JF 1704 | √ | - | - |
| JPR 7150.2 Compliance Matrix | √ | - | - |
| Software Requirements Specification  | - | √ | √ |
| Software Test Plans and Reports | - | √ | √ |
| Version Description Document | - | - | √ |

# SOFTWARE MAINTENANCE/SUSTAINING ENGINEERING PLAN

## MAINTENANCE PROCESS IMPLEMENTATION

The PI’s will be responsible for maintaining the software for the life of the Research Phase. If the software will potentially be used beyond the Research Phase, additional planning may be necessary for flight operational and sustaining activities. This may be in the form of operational and maintenance activities being documented in a Sustaining Engineering Plan and an Operational Transition Plan. :

# SOFTWARE RETIREMENT PLAN

No retirement plan is being provided at this time. The Human Research Program (HRP) will be responsible for determining when the software will be retired.

When it has been determined that the software is no longer required to support operations on ISS, the ISSMP Software Integration Team will be notified via an e-mail that the software is no longer needed.

| Appendix A ACRONYMS AND ABBREVIATIONS |
| --- |
| CCB | Configuration Control Board |
| CM | Configuration Management |
| COTS | Commercial-off-the-shelf |
| CSW | Common Software |
| CWI | Common Work Instruction |
|  |  |
| DR | Discrepancy Report |
|  |  |
| EDS | Electronic Document System |
| ESS | Experiment Support Scientist |
| EST | Experiment Support Team |
| EXPRESS | EXpedite the PRocessing of Experiments to the Space Station |
|  |  |
| FPM | Flight Project Manager |
|  |  |
| GCAR | Government Certification Approval Request |
| GOTS | Government-off-the-shelf |
|  |  |
| HHPC | Human Health and Performance Contract |
| HRF | Human Research Facility |
| HSE | Hardware Support Engineer |
|  |  |
| IDAGS | Integrated Display and Graphics Standard |
| ISS | International Space Station |
| ISSMP | International Space Station Medical Projects |
| IT | Information Technology |
| iV&V | Independent Verification and Validation |
|  |  |
| JF | JSC Form |
| JPR | JSC Procedural Requirement |
| JSC | Johnson Space Center |
| JWI | JSC Work Instruction |
|  |  |
| MIP | Mandatory Inspection Point |
| MOTS | Modified-off-the-shelf |
|  |  |
| NASA | National Aeronautics and Space Administration |
|  |  |
| OTS | Off-the-shelf |
|  |  |
| PAR | Payload Anomaly Report |
| PDRT | Payload Display Review Team |
| PI | Principal Investigator |
| POIF | Payload Operations Integration Function |
| PRCU | Payload Rack Checkout Unit |
| PSCP | Payload Software Control Panel |
| PSE | Payload Support Engineer |
|  |  |
| QARC | Quality Assurance Records Center |
|  |  |
| Rev.  | Revision |
|  |  |
| S/W | Software |
| SA | Human Health and Performance Directorate |
| SATERN | System for Administration, Training, and Educational Resources for NASA |
| SDD | Software Design Document |
| SDP | Software Development Plan |
| SLAMMD | Space Linear Acceleration Mass Measurement Device |
| SQA | Software Quality Assurance |
| SRS | Software Requirement Specification |
| SRVD | System Requirements and Verification Document |
| SVN | Subversion |
|  |  |
| TBD | To be determined |
| TPS | Task Performance Sheet |
|  |  |
| UI | User Interface |
|  |  |
| V&V | Verification and Validation |
| VDD | Version Description Document |
|  |  |
| WI | Work Instruction |

Appendix B JSC Form (JF) 1704

Appendix C NPR 7150.2A Compliance Matrix