AHCA Florida Health Care Connections (FX)

<<Insert Project Name Here>>

Configuration Management Plan

Version: 001

Date: Month Day, YYYY

Author: <Author>

Submitted To: AHCA FX Program Administration Team

Revision History

DATE	VERSION	DESCRIPTION	AUTHOR
M/D/YYYY	001	<<Insert Project Name Here>> Configuration Management Plan first draft version	

Modifications to the approved baseline version (100) of this artifact must be made in accordance with the Artifact Management Standards.

Quality Review History

DATE	REVIEWER	COMMENTS
M/D/YYYY	Name	

Table of Contents

Table of Exhibits

SECTION 1 INTRODUCTION

1.1 BACKGROUND

The Florida Agency for Health Care Administration (AHCA or Agency) is adapting to the changing landscape of healthcare administration and increased use of the Centers for Medicare and Medicaid Services (CMS) Medicaid Information Technology Architecture (MITA) to improve the administration and operation of the Florida Medicaid Enterprise. The current Florida Medicaid Enterprise is complex; it includes services, business processes, data management and processes, technical processes within the Agency, and interconnections and touchpoints with systems necessary for administration of the Florida Medicaid program that reside outside the Agency. The future of the Florida Medicaid Enterprise integration is to allow the Agency to secure services that can interoperate and communicate without relying on a common platform or technology.

The Florida Medicaid Management Information System (FMMIS) has historically been the central system within the Florida Medicaid Enterprise; functioning as the single, integrated system for claims processing and information retrieval. As the Medicaid program has grown more complex, the systems needed to support the Florida Medicaid Enterprise have grown in number and complexity.

The Medicaid Enterprise System (MES) Procurement Project was re-named Florida Health Care Connections (FX) in the summer of 2018. FX is a multi-year transformation to modernize the current Medicaid technology using a modular approach, while simultaneously improving overall Agency functionality and building better connections to other data sources and programs.

1.2 PURPOSE

This Configuration Management (CM) Plan establishes the technical and administrative direction and surveillance for the management of configuration items (i.e., software, hardware, and documentation) associated with the <<Insert Project Name Here>> <(Acronym)> that are to be placed under configuration control. This document defines the project's structure and methods for:

- Identifying, defining, and baselining configuration items (CIs)
- Controlling modifications and releases of CIs
- Reporting and recording status of CIs and any requested modifications
- Ensuring completeness, consistency, and correctness of CIs
- Controlling storage, handling, and delivery of CIs

As the project matures, appropriate sections of this plan will require periodic updating.

1.3 Scope Statement

<Instructions: Provide a high-level overview of the project. Focus on the process and deliverable aspects of the project, including contract type, major milestones, and stakeholders.>

1.3.1 System Description

<Instructions: Provide a brief description of the system, its history, and the environment in which it operates (e.g., mainframe, client/server, stand-alone, etc.).>

1.3.2 System Architecture

<Instructions: Describe the planned system architecture, operating system, and application languages. Identify other legacy or new systems with which this system interfaces.>

1.4 Goals and Objectives

<Instructions: Identify the goals and objectives for this plan.>

- Goal #1 – The goal of this plan is to <insert language>
  - Objective #1 – <insert objective>
  - Objective #2 – <insert objective>
- Goal #2 – The goal of this plan is to <insert language>
  - Objective #1 – <insert objective>
  - Objective #2 – <insert objective>

1.5 Referenced Documents

The following documents were used as input to the development of the CM Plan and provided valuable information to produce the procedures and processes.

- CMS Target Life Cycle (CMS TLC) Configuration Management Plan
- <add additional, as needed>

SECTION 2 ROLES AND RESPONSIBILITIES

Exhibit 2-1: Roles and Responsibilities identify the roles and responsibilities for the primary stakeholders that maintain or use this document.

<Instructions: Specify each major role (not name of the individual) and the major activities related to this document.>

ROLE	RESPONSIBILITY
	▪
	▪
	▪
	▪
	▪
	▪
	▪
	▪

Exhibit 2-1: Roles and Responsibilities

SECTION 3 ASSUMPTIONS/CONSTRAINTS/RISKS

3.1 ASSUMPTIONS

<Instructions: Describe any assumptions or dependencies regarding the CM approach for the project. These may concern such issues as conventions for labeling configuration items, related software or hardware, operating systems, or end-user characteristics.>

3.2 CONSTRAINTS

<Instructions: Describe any limitations or constraints that have a significant impact on configuration management of the project. Such constraints may be imposed by any of the following (the list is not exhaustive):

- Hardware or software environment
- End-user environment
- Availability of resources
- Interoperability requirements
- Interface/protocol requirements
- Data repository and distribution requirements.>

3.3 RISKS

<Instructions: Describe any risks associated with configuration management for the project and proposed mitigation strategies.>

SECTION 4 CONFIGURATION MANAGEMENT APPROACH

4.1 METHODS AND TOOLS

Exhibit 4-1: Configuration Management Methods and Tools describes the methods, processes, tools, and techniques that will be used for configuration management, and how they will integrate with other project processes.

<Instructions: Describe the methods, processes, tools, and techniques that will be used for configuration management, and how they will integrate with other project processes (e.g., change management, quality management, contract management, subcontractor management, project monitoring and control, risk management, etc.). As appropriate, refer to the Project Management Plan (PMP) and/or any applicable subordinate plans that may exist (e.g., Change Management Plan, Quality Management Plan, Release Management Plan, etc.).>

PROCESS	TOOLS & TECHNIQUES
	▪
	▪
	▪

Exhibit 4-1: Configuration Management Methods and Tools

4.2 ENVIRONMENT

<Instructions: Describe the technical architecture, setup, and maintenance of the CM environment.>

SECTION 5 CONFIGURATION MANAGEMENT ADMINISTRATION

5.1 CONFIGURATION IDENTIFICATION

<Instructions: Describe the process for identifying and documenting the functional and physical characteristics of items that are to be placed under configuration control. Identify how CIs will be selected and the types of configuration documentation required for each CI. Also, identify how configuration identification information will be maintained and made readily available to all CMS decision-makers.>

5.2 NAMING STANDARD

<Instructions: Describe the standard to be followed for uniquely naming CIs. For example, naming standards may be based on the type and version of the CI and/or criteria about the component's location, model, function, etc. Describe the hierarchical structure established to identify and summarize the CIs comprising the project, product, or automated system. The identification scheme needs to cover hardware, system software, commercial-off-the-shelf (COTS) products, documentation, and all application development artifacts listed in the product directory structure (e.g., plans, models, software components, test results and data, executables, etc.).>

5.2.1 BASELINES

<Instructions: Identify the types of configuration baselines that will be established for the project. Provide a brief description of what types of CIs each baseline contains. Explain when and how they will be defined. The following are some sample baselines that may be considered for inclusion:

- Functional – Describe where system functional characteristics will be established and the process by which the baseline will be managed.
- Allocated – Describe where the functional and interface characteristics will be established and the process by which the baseline will be managed.
- Development – Describe where the development baseline will be established and the process by which it will be managed.
- Test – Describe where the test baseline will be established and the process by which it will be managed.
- Product – Describe where product baseline (consists of completed and accepted system components and documentation that identifies the products) will be established and the process by which it will be managed.>

5.2.2 STORAGE AND RETENTION

<Instructions: Describe how the CIs are to be retained and recorded in the project inventory (e.g., online, offline, media type, and format). Also describe the overall retention policies of all CIs, especially as they apply to back-ups, contingency plans, and disaster recovery plans.>

5.3 CONFIGURATION CONTROL

<Instructions: Identify when CIs will be placed under configuration control. Identify and briefly describe the control mechanisms that will be used for establishing CM baselines and approving or disapproving subsequent changes to those baselines (e.g., Configuration (or Change) Control Board (CCB)).>

5.3.1 IMPACT ANALYSIS

<Instructions: Define the process by which the impact of approved changes and/or problem reports affecting baselined CIs under configuration control are assessed.>

5.3.2 TRACKING AND CONTROLLING CHANGES TO CI BASELINE

<Instructions: Describe how approved changes to identified CIs and/or changes to the CI List are developed or maintained by the project will be properly documented, implemented, verified, and tracked to ensure incorporation in all applicable systems and/or products. If applicable, describe how changes identified during ongoing maintenance of products/systems operating in production will cycle forward into new business needs for appropriate analysis and consideration prior to modification of existing, or development of new, products/systems in response to requested changes.>

5.4 CONFIGURATION INTEGRITY

<Instructions: Describe the process used to build managed and controlled baselines. A flow diagram and narrative of the progress of CIs through the CM life cycle is recommended. Documentation baselines should be noted. Describe how the versions of each CI in a baseline will be tracked, and how the differences between baselines will be determined and reported. Identify how audits of software and hardware configuration baselines in the production environment will be performed to ascertain that no unauthorized changes have been made without proper approval.>

5.4.1 CONFIGURATION STATUS ACCOUNTING

<Instructions: Describe the process for recording and reporting information needed to maintain integrity and traceability of controlled CIs and associated documentation throughout the life cycle. This includes the process for monitoring the status of proposed changes and the implementation status of approved changes (e.g., what changes have been made, when the changes were made, and what components were affected by the changes). Examples of CM records might include a Change Request Log, a CI Revision History, etc. Describe how site configuration data will be developed and maintained and plans for the incorporation of modification data on products and CIs. Also, identify how configuration status accounting information will be maintained and made readily available to all CMS decision-makers.>

5.4.2 CONFIGURATION AUDITS

<Instructions: Describe the process for performing internal and external configuration audits of identified and controlled CIs (software and non-software). Specify the type and number of audits to be conducted during the project life cycle. The level of rigor or frequency of audits should be adjusted according to the size and complexity of the project. For developmental and operational systems, describe the process for periodically reconciling against their documentation to ensure consistency between the product and its current baseline documentation.>

5.4.3 CONTROL CHANGES

<Instructions: Describe the process for changes and controls.>

5.4.4 AUDIT CHANGES

<Instructions: Describe the process for audit changes.>

5.4.5 BUILD ARTIFACT RETENTION

<Instructions: Describe the process for Build artifact retention.>

SECTION 6 ENVIRONMENT PLAN

<Instructions: Describe the different available environments for reference to the Release and Configuration Management Plan. This section does not supersede, replace, or otherwise modify other deliverables related to environment specifications, planning, purpose, etc.>

6.1 ENVIRONMENT LIST

<Instructions: Describe the environments planned for the FX. This includes the owner of the environment, the data strategy for the environment, whether the environment is highly available (multi-node), and a description of the activities to be performed in the environment.>

Exhibit 6-1: Environment List describes the environments planned for the FX.

ENVIRONMENT	ATTRIBUTES	DESCRIPTION/PURPOSE
	▪	

Exhibit 6-1: Environment List

Exhibit 6-2: Access Matrix displays the access matrix.

ENVIRONMENT	SYSTEM ADMINISTRATION ACCESS (ADMIN CONSOLES)	APPLICATION CONFIGURATION ACCESS	APPLICATION DATA ACCESS
	▪	▪	▪

Exhibit 6-2: Access Matrix

6.2 PROMOTION PATHS

<Instructions: Describe the promotion path for this development effort to get to the target Production Environment. Also describe the emergency pathway for isolated critical break-fix items that are deemed requiring an expedited timeline/path to Production. Additionally, this section displays the ALM tools promotion process.>

6.2.1 RELEASE PROMOTION PATH

<Instructions: Describe the Release Promotion Path.>

6.2.2 EMERGENCY/MINOR RELEASE PROMOTION PATH

< Instructions: Describe the Emergency or Minor Release path. This emergency pathway would be for a change that was deemed low risk, low impact, and/or business/operational priority warranting the use of the Emergency Branch and release process.>

6.2.3 ALM PROMOTION PATH

<Instructions: Describe the ALM promotion path.>

SECTION 7 RELEASE MANAGEMENT PLAN

<Instructions: Describe the Release Management Plan.>

7.1 SCOPE DETERMINATION/IDENTIFICATION

<Instructions: Describe Scope Determination/Identification. Items included in the Release Plan should support the release goals and create a target build artifact that can be used to improve the business state or operations in production.>

7.2 SCHEDULING A RELEASE

<Instructions: Describe Scheduling a Release. The release schedule includes not only a production deployment date but also interim milestones like testing completion and development completion dates.>

7.3 RELEASE APPROVAL PROCESS

<Instructions: Describe the appropriate Release approval process. Include Major, Minor, and ALM.>

7.3.1 MAJOR RELEASE APPROVAL PROCESS FLOW

<Instructions: Describe the process flow for a Major Release.>

7.3.2 MINOR RELEASE APPROVAL PROCESS FLOW

<Instructions: Describe the process flow for a Minor Release.>

7.3.3 ALM RELEASE APPROVAL PROCESS FLOW

Instructions: In addition to configuration releases, there is also occasionally a need to change attributes, add work item types, or modify workflows in the ALM toolset supporting the efforts of the project. Describe the ALM approval workflow.>

APPENDICES