DEPARTAMENTO DE
SALUD

Deliverable 213-HLD_Hacienda Batch v2.2: Puerto Rico Department of Treasury (Hacienda) Batch Participant Information Interface

MEDITI 3G Project
Government of Puerto Rico

Version 2.2
September 22, 2023

Contract #2022-DS0707-E

Document Revision History

Version Number	Date	Description
0.1	04/20/21	Initial Draft.
0.2	04/27/21	Internal Review.
1.0	04/28/21	Submission for approval.
2.0	03/31/2022	• Updated document deliverable number from 98 to 213-HLD V1.0 • Updated PREE to MEDITI3G where applicable • Updated Pub-1075 for MARS-E (if applicable) • Updated 2.4 Citations (removing Pub-1075 if applicable) • Updated 2.5 Stakeholders (remove NTT Data) • Removed references to PHI and FTI
2.1	7/26/22	• Updates to the messaging schemas and elements
2.2	9/22/2023	Updated sections: 3.1.6.2, 3.1.4.1, 3.3.2.4.5, 5.2

Document Approval

Stakeholder Name	Stakeholder Role	Stakeholder Signature	Signature Date (MM/DD/YYYY)
Alexander Quevedo	State HIT Coordinator		

Send inquiries to:
Wovenware
1000 Los Angeles Street Suite 100
San Juan, PR 00909

Email:
mediti3g@wovenware.com

Table of Contents

This document is intended for internal use only.

LIST OF TABLES

LIST OF FIGURES

1 Acronyms

Table 1 - List of Acronyms

Acronym/Term	Definition
AES	Advanced Encryption Standard
AES 256	Advanced Encryption Standard with a key size of 256 bits
BA	Business Architecture
CMS	Centers for Medicare and Medicaid Services
ESB	Enterprise Service Bus
GB	Giga Bytes
GHP	Government Health Plan
Hacienda	Spanish name for Puerto Rico Department of Treasury
HIPAA	Health Insurance Portability and Accountability Act of 1996
HIT	Health Information Technology
HIPS	Host Intrusion Prevention System. A HIPS is a system or program designed to protect a computer system from virus and malware.
HITECH Act	Health Information Technology for Economic and Clinical Health Act
HTTPS	Hypertext Transfer Protocol Secure
IA	Information Architecture
ID	Identification
IT	Information Technology
IOPS	input/output operations per second
KB	Kilo Bytes
LAN	Local Area Network
LRS	Locally Redundant Storage
MARS-E	Minimum Acceptable Risk Standards for Exchanges
MEDITI2	Medicaid Integration Technology Initiative 2
MEDITI3G	Medicaid Integration Technology Initially 3rd Generation 3
MB	Mega Bytes
NIEM	National Information Exchange Model
OS	Operating System
PII	Personally Identifiable Information
P.R.	Puerto Rico
PRDoH	Puerto Rico Department of Health
PREE	Puerto Rico Eligibility and Enrollment. Original Project name.

PRMP	Puerto Rico Medicaid Program
RAM	Random Access Memory
REST	Representational State Transfer
SI	System Integrator
SFTP	Secure File Transfer Protocol
SLA	Service Level Agreement
SOA	Service Oriented Architecture
SQL	Structured Query Language
SSD	Solid State Drive
SSH-2	Secure Shell Version 2
SSL	Secure Socket Layer
State Hub	State Data Verification Hub
TA	Technical Architecture
TDS	Trusted Data Source
UML	Unified Modeling Language
U.S.	United States
vCPU	Virtual Central Processing Unit
VPN	Virtual Private Network
WAN	Wide Area Network
XML	eXtensible Markup Language
XSD	XML Schema Definition

2 Introduction

The following document shall provide a high-level design of the batch interaction in a State Hub Data Verification Interface between a Requestor System and the Puerto Rico Department of Treasury or Hacienda, an agency that handles income and non-MAGI (non-Modified Adjusted Gross Income) information supplied by applicants in their tax returns.

The Hacienda Batch Participant Information Interface (henceforth Local Interface) shall allow a Requestor System to send a batch request file to gather personal, address, income, and non-MAGI information about participants in the Hacienda System. The Local Interface shall collect the information returned by Hacienda (henceforth Local Agency) and return it to the Requestor System.

2.1 Objective

The purpose of this document is to provide a high-level design for the Local Interface related to PRMP's MEDITI3G Project based upon the approved document titled PREE Requirements and Definitions: State Data Verification Hub Requirements (Version 1.3). It provides a high-level design view of the Local Interface for the project's stakeholders and, more specifically, the developers involved.

*Note: PREE was the previous project name. Only document titles throughout this document have not been changed from PREE to MEDITI3G.

2.2 Document Scope

This document is intended to cover at a high-level the architecture and design aspects of the Hacienda Batch Participant Information Interface that shall be provided as part of the MEDITI3G Project solution. Software architecture and software design are two important parts of software development. The architecture presented in this document provides a conceptual and logical representation of the Local Interface, capable of complying with the State Hub requirements and expectations already approved. The design focuses on providing a concrete view on how the architecture shall be satisfied by means of vendor specific technologies and custom fit development.

2.3 Intended Audience

The intended audience for this document is all technical personnel involved in the MEDITI3G Project.

2.4 Citations

The high-level architecture and design of the Local Interface is based on the requirements, regulations, conditions, and guidelines established in the following documents:

1. Wovenware's active contract found in the Puerto Rico Comptroller's office.
2. PREE Requirements and Definitions: State Data Verification Hub Requirements (Version 1.3; approved on 8/29/19). Document Tracking Log ID: EE-DT00234
3. MARS-E 2.2 (Volume I: Harmonized Security and Privacy Framework)
4. HIPAA Privacy Rule (Summary of the HIPPA Privacy Rule)
5. HIPAA Security Rule (The Security Rule)
6. HITECH Act Enforcement Interim Final Rule

   (https://www.congress.gov/bill/111th-congress/house-bill/1/text)
7. CMS Seven Standards and Conditions

   (https://www.regulations.gov/document/CMS-2010-0251-0002)
8. NIEM 2.0 (http://niem.github.io/niem-releases/)
9. Deliverable 213: Hacienda Batch Participant Information ICD v2.0 (Found in Project's DDI SharePoint Site)
10. Wovenware System Security Plan
11. Deliverable 204 - State Data Verification Hub High Level Design

2.5 Stakeholders

For purposes of this document's scope, the list of stakeholders is as follows:

Table 2 – MEDITI 3G Project Stakeholders

Stakeholder Name	Role
Puerto Rico Medicaid Program (PRMP)	Sponsor
Puerto Rico Department of Health (PRDoH)	Sponsor
Intervoice	Project Management Office (PMO)
Hacienda	Local Government Agency
RedMane	SI
Wovenware	SI
BerryDunn	OBC Consultant

3 High-Level Architecture

This section covers the high-level architecture for the Local Interface. It contains three (3) interrelated architectures: Business Architecture, Information Architecture, and Technical Architecture. The business capabilities from the Business Architecture define the data strategy of the Information Architecture and design the business services and technical modules of the Technical Architecture.

3.1 Business Architecture

This section covers the Business Architecture for the Local Interface. It defines the business processes and capabilities the Local Interface offers, per approved functional requirements. In addition, it includes business factors that may influence the technical design; factors such as assumptions, constraints, and goals. Finally, it provides information about the anticipated volume of transactions and performance expectations.

3.1.1 Background

For details, refer to the document Deliverable 204 - State Hub High-Level Document, Section 3.1.1 Background. This document is found in MEDITI3G's DDI SharePoint Site.

3.1.2 Proposed System

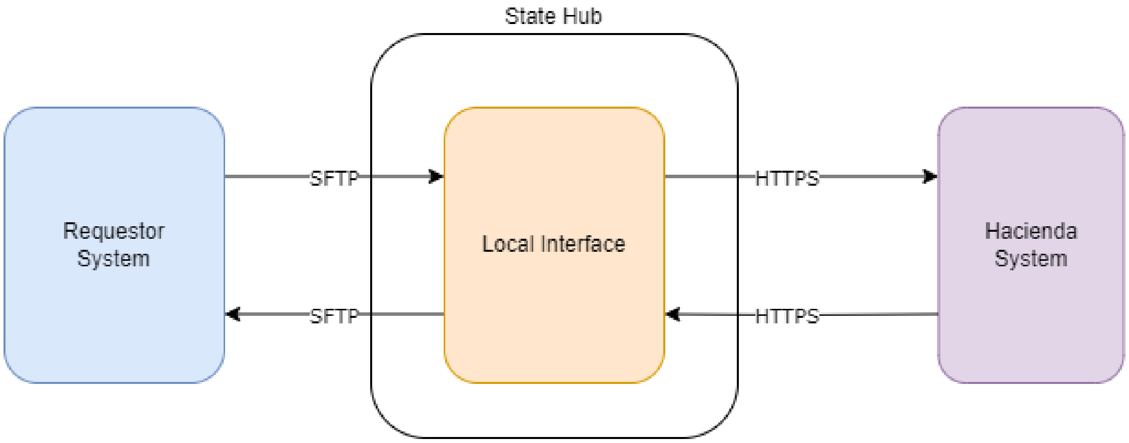
A Local Interface shall be created to exchange information between the Requestor System and the Hacienda System. The interface receives the batch requests that shall contain a participant basic personally identifiable information. The Local Interface shall query the Local Agency System to find the information in their system. The interface shall return the responses to the Requestor System thought the State Hub.

This proposed system shall gather the requested information for the Requestor System. The information helps PRMP to determine the eligibility of a participant automatically. This reduces the population that shall have to visit the PRMP offices for renewal of eligibility for PRMP benefits.

This solution establishes that the Local Interface be implemented as core components of the State Hub in an Azure Government environment to guarantee high availability, redundancy, data integrity and data security using the Minimum Acceptable Risk Standards for Exchanges (MARS-E), Health Insurance Portability and Accountability Act of 1996 (HIPAA) Privacy Rule, HIPAA Security Rule, and CMS Standards and Conditions as the basis.

The following diagram represents the proposed solution from a high-level perspective.

Figure 1 - Proposed Solution (To-Be)



3.1.3 Operational Requirements and Characteristics

The Local Interface shall provide capabilities for a Secure File Transfer Protocol (SFTP) connection, receive batch files, process batch files, deliver the batch response, provide a Negative Acknowledgement (NACK), and connect to the Local Agency via HTTPS requests.

The Local Interface shall be hosted in the State Data Verification Hub (State Hub).

3.1.4 Business Processes Supported

All PRMP beneficiaries (approximately 1.3 million) are required to renew their eligibility. To achieve this, PRMP manages approximately up to three hundred thousand (300,000) renewals monthly. As part of the MEDITI3G solution, these monthly renewals shall be managed through batch requests and batch responses directly between local agencies and the Requestor System to acquire the necessary information for the PRMP eligibility determination.

The Local Interface shall be able to handle several scenarios:

1. Process Batch Request
2. Maximum Response Time Reached
3. Trusted Data Source (TDS) Unreachable
4. Twenty percent (20%) or more errors
5. Delete Files in Inbound Folder
6. Delete Files in Outbound Folder
7. Enable/Disable the Interface

These scenarios shall be explained in the following subsections.

3.1.4.1 Process Batch Requests

For this scenario, the Requestor System sends a batch request to the State Hub specifying the evidence data required for each member and from which local agency the information shall be required. The State Hub shall receive the batch request in a ZIP file via the State Hub's SFTP Inbound Folder and trigger the execution of the Local Interface.

The requests received are validated against their Extensible Markup Language Schema Definition (XSD) to make sure they follow National Information Exchange Model (NIEM) 2.0 standards. After the NIEM validation is made on the manifest and request file, the Local Interface shall perform metadata validations following the information provided in the manifest file.

Once the Local Interface has validated the batch request, it shall start processing the requests by sending individual requests to the local agency in reference. The local agency shall process the request and store the response file in the Outbound folder of the State Hub SFTP. If no information is found, or some other error condition occurs on the Local Agency, the response shall include an error message.

The Local Interface shall split the individual responses of the response file in different response files of up to 100MB. Then, it shall create a manifest and store a ZIP file containing both in the Outbound folder of the State Hub SFTP for the Requestor System to retrieve it.

The Local Interface shall create a NACK file when a request or response file is not valid, when an uncompressed Extensible Markup Language (XML) request file exceeds the 250MB limit, and when the interface was not able to communicate with the Local Agency within the maximum response time provided and no partial response has been collected. After the NACK is created, the interface shall deliver it to the designated Outbound folder.

The responses received are validated against their XSD's. If the responses are not valid, the process shall generate a NACK.

After the Local Interface has validated the response received from the Local Agency and the response manifest file created, both files are compressed in a ZIP file and deposited in the Requester's Outbound Folder.

3.1.4.2 Maximum Response Time Reached

For this scenario, the Local Interface shall monitor that the execution of the batch transaction does not exceed the maximum response time. The maximum response time is a modifiable system administration configuration with an initial default setting of nine (9) calendar days for batch transactions and eighteen (18) calendar days for high availability disaster recovery (HADR). If the maximum execution time is reached and there have been no partial responses collected, the Local Interface shall return a NACK to the Requestor System. If there have been partial results collected during the execution, the responses are validated, a manifest file is created, the files are compressed in a ZIP file and delivered to the Requestor System in their designated Outbound Folder.

3.1.4.3 TDS Unreachable

For this scenario, the Local Interface shall continue to make requests every thirty (30) minutes up to the nine (9) calendar days since the initial request. If the communication between the agency and the Local Interface is failing for twenty-four (24) hours consecutively, the State Hub shall stop processing the transaction. If there are responses from the agency, the requestor system shall receive a partial response. If no responses obtained, the requestor system shall receive a NACK.

3.1.4.4 Twenty percent (20%) or more errors

For this scenario, the Local Interface has been executing but has received twenty percent (20%) sequential errors. In this case, the Local Interface shall stop execution and validate the partial responses. If there are no validation errors on the partial responses, the manifest is created and both files are compressed in a ZIP file to be delivered to the Requester. If there are no partial responses, or there is a validation error on the partial responses, a NACK is returned to the Requester.

3.1.4.5 Delete Files in Inbound Folder

The service adapter that processes inbound files is responsible for deleting the files after processing them. However, as a safety measure, the Blob Storage Lifecycle of the SFTP Server shall be configured to delete any file left more than fourteen (14) days.

When a file is successfully deleted from the Storage Account, it is immediately removed from the storage account's index and is no longer accessible to clients. The blob's data is later removed from the service during Microsoft's garbage collection process. The life cycle configuration is part of the Configuration Management section in the System Security Plan Document.

The “Enable soft delete for blobs” is disabled for blobs and containers. This means that the information is purged without the possibility of being recovered. There are no Time-based retention policies for immutable blob data. If the policy is created for the objects, the minimum you can configure the policy is for 1 day. This would interfere with the application that is set to delete the request file after it processes it. The requester is responsible for deleting the response file, if not, the deletion policy shall delete it.

The Azure Resource Manager lock is a configuration that allows you to lock a subscription, resource group, or resource to prevent accidental or malicious deletion or modification of resources. Applying the lock to the storage account does not protect containers or blobs within that account from being deleted or overwritten.

Network access to the Storage Account (SA) is done via virtual network (public access to the Storage Account is disabled). Public access to blobs (anonymous) is disabled. Minimum Transport Layer Security version is set to 1.2.

3.1.4.6 Delete Files in Outbound Folder

The service adapter that processes outbound files is the responsible for deleting the files after processing them. However, as a safety measure, the Blob Storage Lifecycle of the State Hub Blob Storage shall be configured to delete any file left more than fourteen (14) days.

When a file is successfully deleted from the Storage Account, it is immediately removed from the storage account's index and is no longer accessible to clients. The blob's data is later removed from the service during Microsoft's garbage collection process. The life cycle configuration is part of the Configuration Management section in the System Security Plan Document.

The “Enable soft delete for blobs” is disabled for blobs and containers. This means that the information is purged without the possibility of being recovered. There are no Time-based retention policies for immutable blob data. If the policy is created for the objects, the minimum you can configure the policy is for 1 day. This would interfere with the application that is set to delete the request file after it processes it. The requester is responsible for deleting the response file, if not, the deletion policy shall delete it.

The Azure Resource Manager lock is a configuration that allows you to lock a subscription, resource group, or resource to prevent accidental or malicious deletion or modification of resources. Applying the lock to storage account does not protect containers or blobs within that account from being deleted or overwritten.

Network access to the Storage Account is done via virtual network (public access to the Storage Account is disabled). Public access to blobs (anonymous) is disabled. Minimum Transport Layer Security version is set to 1.2.

3.1.4.7 Enable/Disable the Interface

For this scenario, the State Hub administrator shall have the ability to enable and disable the availability of the Local Interface.

3.1.5 High Level Functional Requirements

The Local Interface, as part of the State Data Verification Hub, shall help in the determination process of eligibility by including functional capabilities. A summary of the functional requirements is presented below. Refer to section 5.1 for the complete list of functional requirements.

1. Support for logging of events and preventing their manipulation, in accordance with MARS-E Audit control Family requirements.
    a. The only user allowed to manipulate logs shall be the State Hub Administrator. All other users shall have read-only access to the logs.
2. Support for an effective batch information exchange with service providers.
3. Compliance with CMS Conditions and Standards.
    a. Modularity Standard - The Local Interface is being developed using system development lifecycle methodology for improved efficiency and quality of service (communication between PRMP and the local agency).
    b. MITA Condition - the workflows are designed to support the operation of the P.R. Medicaid program and the exchange of data with the local agency.
    c. Industry Standards Condition - The Local Interfaces supports NIEM 2.0 and common Internet communication protocols.
    d. Leverage Condition - The Local Interface has been developed to use the Microsoft Azure Government Cloud, which consists of technologies already developed, consistent with a service-oriented architecture, from commercially products.
    e. Business Results Condition - The Local Interface shall support accurate and timely processing of claims (including claims of eligibility), adjudications, and effective communications with the local agencies (data providers), by requesting and receiving the agency's beneficiary data.
    f. Reporting Condition - The local interface shall produce and capture (log) transaction data that shall be used to generate reports, and capture information that shall contribute to program evaluation and continuous improvement in business operations.
    g. Interoperability Condition - the Local Interface supports communication between PRMP and the local agency. This promotes effective citizen service and better management and health services to beneficiaries.
4. Support for the protection of Personally Identifiable Information (PII)
5. Compliance with security standards and regulations.

6. Compliance with the applicable rules and regulation of the MARS-E 2.2 framework.

3.1.6 Factors Influencing Technical Design

There are several factors that influence the technical design of the Local Interface. They have been categorized as assumptions, constraints, risks, and design considerations.

3.1.6.1 Assumptions

The following assumptions have been identified:

1. Azure Government shall maintain backwards compatibility for up to three (3) versions allowing enough time to update code for new offerings of services and components. The inclusion of new offerings later shall not negatively impact compatibility and compliance with HIPAA, and MARS-E.
2. The Requestor System shall use the interface to assist Medicaid in determining renewal eligibility of PRMP participants.
3. The identified MEDITI3G operational personnel shall establish the necessary procedures to grant access to the SFTP Server.
4. The Requestor System shall request the Local Agency through the Local Interface only.
5. The Local Agency shall promptly notify the identified MEDITI3G operational personnel of any maintenance window not previously scheduled or agreed upon.
6. The participant's SSN has been pre-validated prior to a local interface match being requested for the Requestor System. Pre-validation shall be performed by PRMP manually or electronically.
7. Hacienda shall provide an HTTPS REST endpoint to process the requests individually.
8. The Hacienda System shall support up to two thousand (2,000) parallel connections.
9. The Local Interface shall contain the data elements from the years 2020, 2021, and 2022.
10. If there are no communication problems, the Hacienda Batch Environment shall be available 24 hours a day, 7 days a week, except for previously scheduled and notified downtimes and maintenance windows.

11. The Hacienda Batch Environment shall accept SSN, date of birth, and full name (First Name and Surname) as the request's parameters.
12. Hacienda's endpoints shall provide the most up-to-date information available.
13. Batch Local Interface shall handle a maximum of three hundred thousand (300,000) transactions per month.
14. The TDS response shall include the updated date of the information returned as the replicated timestamp.
15. Hacienda's real-time endpoint shall be available in a timely fashion unless otherwise notified to PRDoH's identified MEDITI3G operational personnel.
16. Hacienda shall implement the necessary normalization rules to allow the requests to be sent without special characters and match the records.
17. The Hacienda Batch Interface shall be backwards compatible with future changes with the Hacienda System data.

3.1.6.2 Constraints

The constraints below have been identified:

1. The interface shall be dedicated to connecting to a single Trusted Data Source (TDS) for requesting data.
2. The State Hub, the environment that shall contain the Local Interface, shall not manage files greater than 100 GB.
3. The Local Interface shall perform searches based on the combination given of SSN, Name (First Name and Last Name), and DOB. Since all fields are needed, the ability to find a result with wrong information or miswritten values depends entirely on the matching algorithms of the Local Agency, if any.
4. If for some reason the Local Agency has both Last Name (Surname) and Second Last Name (Maiden name) in the same field, the Requestor System shall need to merge both last names in the request.
5. The maximum response time for batch shall be nine (9) calendar days.
6. The maximum number of concurrent batch requests shall be one (1).
7. Each batch request shall have its ZIP file.
8. The XML request file within the ZIP file shall not be greater than 250MB.
9. The Federal Hub implements NIEM 2.0 and has not indicated when they would upgrade. Since newer versions are not backward compatible with older versions, the State Hub and the Local Interfaces shall also use NIEM 2.0.
10. The Hacienda System supports individual requests.

11. The Hacienda System requires that the requester information is sent as part of the request to their system.

3.1.6.3 Risks

The following section lists the risks or issues that are currently open in the project's SharePoint site: PREE DDI - Home (sharepoint.com):

1. EE-RI00361: Hacienda rejects proposed layout in ICD.

3.1.6.4 Issues

No risks nor issues are currently open in the project's SharePoint site: PREE DDI - Home (sharepoint.com)

3.1.6.5 Design Considerations

The design of the Local Interface shall keep in mind the following considerations:

1. Establish a long-term data hub for PRDoH with the Local Interface operating within the hub.
2. To enhance any current or future needs (such as scalability), as relevant for the Local Interface.
3. Adhere to Service Oriented Architecture (SOA) best practices.
4. Maximize simplicity by using Commercial off-the-shell (COTS) technologies whenever possible over custom work.
5. The identified MEDITI3G key personnel shall be responsible for establishing all operational procedures.
6. Detailed information regarding Shared Responsibility Security Matrix outlining the security requirements established in MARS-E shall be addressed in the SSP document. This includes topics such as Audit requirements, Configuration management, References to Release Management or change control via online access to the portal, and so forth.

Note: Although, ESB and SOA were considered during the design phase, they shall not be implemented at this time.

3.1.7 Anticipated Volume and Performance Expectations

For details, see the document Deliverable 204 - State Hub High-Level Document in the Section 3.1.8 Anticipated Volume and Performance Expectations. This document is found in the MEDITI3G SharePoint site.

3.2 Information Architecture

This section covers the Information Architecture for the Local Interface. It defines the data models and standards of the data for the exchange of information between the interface, requesters, and the Local Agencies.

3.2.1 Conceptual Data Model

For details, see the document Deliverable 213: Hacienda Batch ICD in the Section 5.1.5 Message Format (or Record Layout) and Required Protocols.

3.2.2 Data Management Strategy

This section covers the data management strategy that governs the Local Interface data members. It describes how the data comes into the system, may get modified, retrieved, and subsequently removed from the system.

3.2.2.1 Request Manifest

The Request Manifest data is submitted by the Requester via the State Hub's SFTP. The data shall be submitted inside a ZIP file and placed in the designated Inbound folder in the SFTP. The ZIP file is decompressed when the Local Interface detects that the file was placed in the Inbound Folder. After the decompression, the request manifest is retrieved in XML format following NIEM standard. The data in transit is not modified. After the data is used, it is deleted from the Local Interface. Only metadata and other audit related information is captured from the request manifest to add to audit logs. As soon as the interface picks up the file and validates it, the ZIP file is deleted.

3.2.2.2 Response Manifest

The Response Manifest is generated by the Local Interface when valid responses are received from the Local Agency. The file is created by the State Hub in XML format following NIEM standards, containing metadata information on the response that is returned with the Response Manifest. After the creation, the file is compressed in a ZIP file with the response and is returned to the Requester to its designated Outbound Folder in the State Hub's SFTP folder. When in transit, the file is not modified. The ZIP file contains PII data in the agency request file that is not persisted.

3.2.2.3 Local Agency Request

The Local Agency request data is submitted by the Requester via the State Hub's SFTP. The data shall be submitted inside a ZIP file and placed in the designated Inbound folder in the SFTP. The ZIP file is decompressed when the Local Interface detects that the file was placed in the Inbound Folder. After the decompression, the request data is retrieved in XML format following NIEM standard. The data in transit is not modified and after the data is used it is deleted from the Local Interface. The ZIP file contains PII data in the agency request file, but no PII data is persisted. As soon as the interface picks up the file and validates it, the ZIP file is deleted and the response data is sent to the SFTP Server, so the Local Agency shall retrieve the information.

3.2.2.4 Local Agency Response

The Local Agency response is created by the Local Interface when valid responses are received from the Local Agency. The file is created by the State Hub in XML format following NIEM standards, containing the information on the response that is returned from the Local Agency. After the creation, the file is compressed in a ZIP file with the response manifest and is returned to the Requester to its designated Outbound Folder in the State Hub's SFTP folder. When in transit, the file is not modified. The ZIP file contains PII data in the agency response file that is not persisted.

3.2.3 Messaging and Standards Support

The messaging standards supported are:

1. SSH-2
2. XML
3. NIEM 2.0
4. ZIP file format
5. JSON
6. REST

3.3 Technical Architecture

This section covers the Technical Architecture for the Local Interface. It describes the technical and application design aspects by leveraging industry standards and best practices. It focuses in providing a vendor independent conceptual and logical view of the Local Interface.

3.3.1 Enterprise Architecture

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.1 Enterprise Architecture. This document is found in the MEDITI3G SharePoint site.

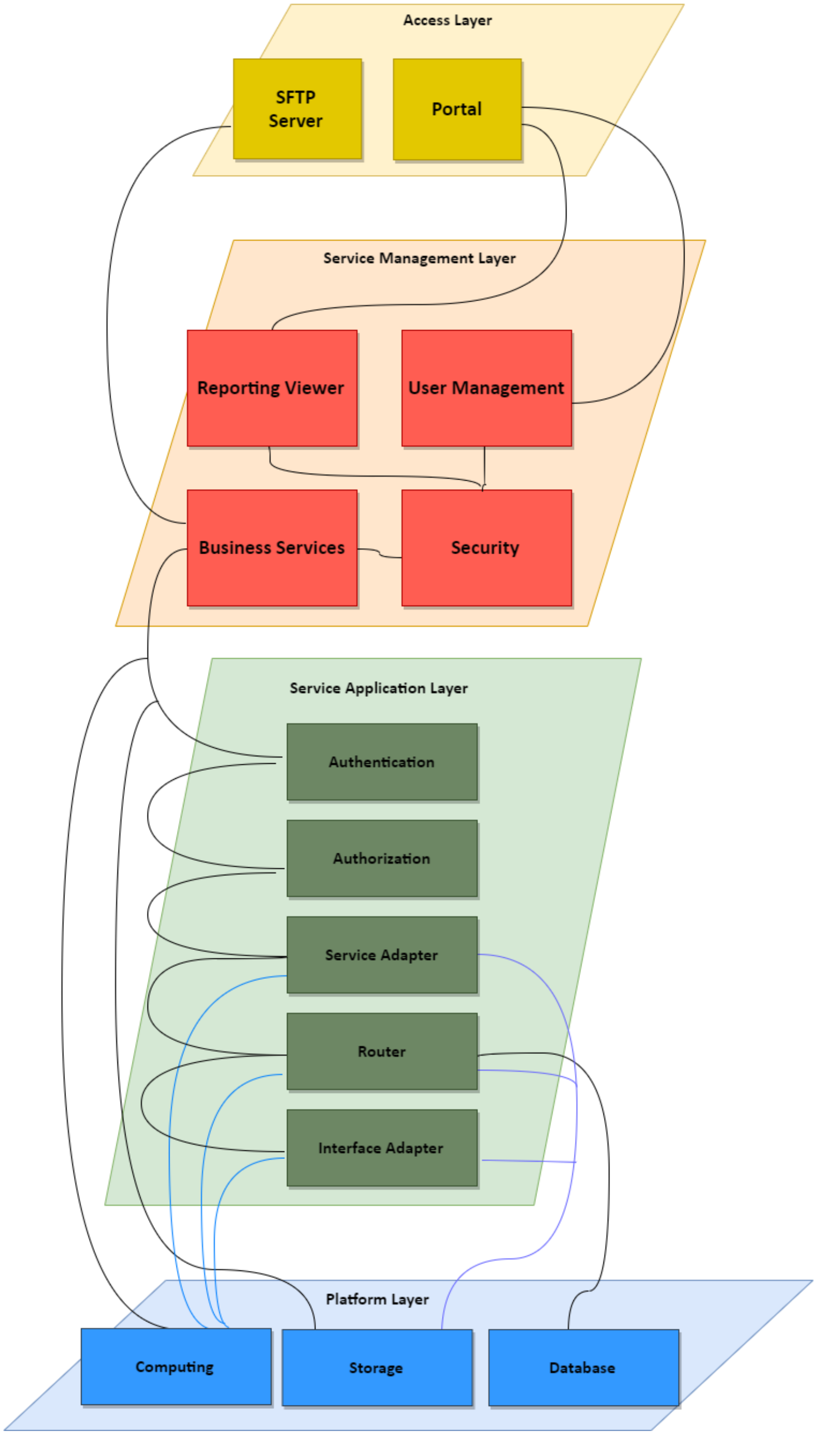
3.3.2 Application Architecture

For details, see the document State Hub High-Level Document in the Section 3.3.2 Application Architecture.

3.3.2.1 Composition Model

The multilayer Application Architecture represents the combination of applications and connections to deliver the services for the Local Interface Requesters. Figure 2 - Multilayer Application Architecture Model illustrates the relationship between the layers and their components.

Figure 2 - Multilayer Application Architecture Model



3.3.2.2 Access Layer

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.1 Access Layer. This document is found in the MEDITI3G Project's SharePoint site.

3.3.2.2.1 SFTP Server

For details, refer the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.1.2 SFTP Server. This document is found in the MEDITI3G Project's SharePoint site.

3.3.2.2.2 Portal

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.1.3 Portal. This document is found in the MEDITI3G Project's SharePoint site.

3.3.2.3 Service Management Layer

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.2 Service Management Layer. This document is found in the MEDITI3G Project's SharePoint site.

3.3.2.3.1 Business Service Module

The business services describe the SOA services which perform specific business needs. The goal of the business services module is to allow for interoperable business processes amongst disparate entities. The business services are used as interface adapters that enable the communication between the State Hub and the local agency.

The solution shall leverage Service Adapters as the primary module for realizing SOA capabilities. A service-oriented architecture provides services that have definitive business value to the PRMP, such as calls to local agencies to verify identity and citizenship of individuals requesting medical assistance and to verify individual income. The intent is to leverage or extend services that currently exist, and for new services it is expected that services shall adhere to common business service definitions.

It is also responsible for adhering to the principle of "Fail Safe" to ensure that a system in a failed state does not reveal any sensitive information or leave any access controls open for attacks.

The services in this module may also distribute a message to multiple destinations based on a message attribute or service provider availability through sequential, parallel, and aggregated choreography and orchestration.

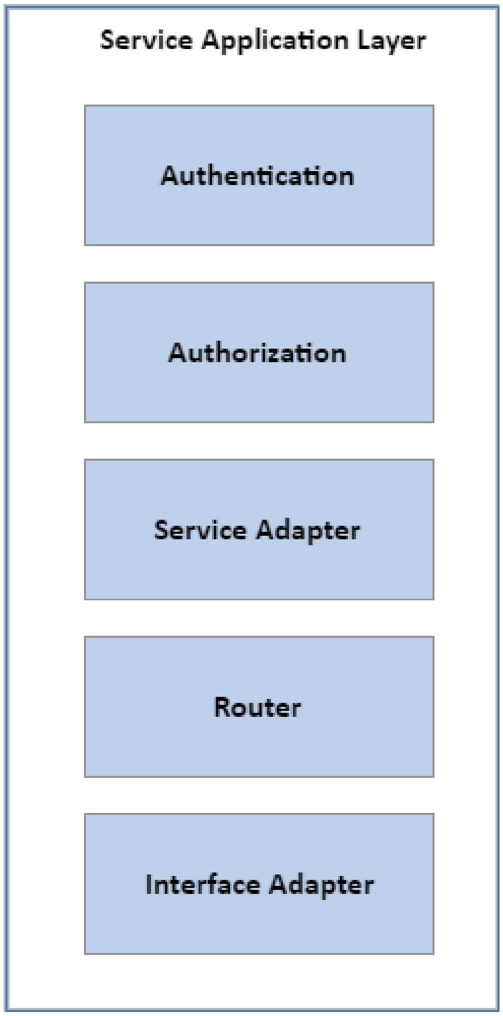
3.3.2.3.2 Security Module

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.2.6 Security Module. This document is found in the MEDITI3G Project’s SharePoint site.

3.3.2.4 Service Application Layer

The Service Application Layer consists of the modules a Service Requestor uses when determining a participant’s eligibility. The modules that are part of this layer oversee the receiving, handling, and routing of the requests. The business logic portion of the solution shall reside in this layer and accesses the necessary technical services to complete the request life cycle.

Figure 3 - Service Application Layer



3.3.2.4.1 Authentication Module

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.3.7 Authentication Module. This document is found in the MEDITI3G Project's SharePoint site.

3.3.2.4.2 Authorization Module

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.3.7 Authorization Module. This document is found in the MEDITI3G Project's SharePoint site.

3.3.2.4.3 Service Adapter Module

The Service Adapter Module is a service wrapper that encapsulates the service exposed by the Local Interface. The adapter supports incoming and outgoing real-time requests and responses to and from one or many service providers, as well as batch requests to and responses from the service providers.

The Service Adapter continues to accept requests and send responses, despite having some or all its service providers off-line. It provides the capability of enabling or disabling services (interfaces) on demand. Even if the agency is down or if it cannot produce a response, the local interface shall respond indicating that an error has occurred.

The module implements an Application-to-Application behavior, supporting asynchronous message correlation so that the target's response is associated with the appropriate request made by the source.

The module shall:

1. Process all PII dynamically to not retain this data permanently.
2. Provide a reliable, once-only delivery of messages, meaning that requests and responses are non-repetitive.
3. Receive messages and respond back to the requester even when timeouts occur within the Local Interface.
4. Provide downstream throttling in scenarios where downstream usage is such that it guarantees delivery of the responses back to the requester, the downstream usage needs to be reduced temporarily.
5. Transform incoming and outgoing messages according to their destination's messaging capabilities and perform source to destination file integrity checks for exchange of data.

3.3.2.4.4 Router Module

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.3.4 Router Module. This document is found in the MEDITI3G Project’s SharePoint site.

3.3.2.4.5 Interface Adapter Module

The Interface Adapter Module is the service wrapper in charge of processing data. The module provides support messages and data formats based on logical representations of business objects rather than native application data structures. This module shall process up to one (1) concurrent batch request file.

Message validations for requests and responses are performed to ensure the payloads are properly constructed in compliance with the defined protocols (see Citations Section).

The module is responsible for querying the local agency System for Participant(s) Information, to be returned within one or more response files. The request file shall contain a batch set of individual participant requests, each request containing participants PII search criteria.

Message validations for requests and responses ensure the payloads are properly constructed in compliance with NIEM 2.0.

3.3.2.5 Platform Layer

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.4 Platform Layer. This document is found in the MEDITI3G Project’s SharePoint site.

3.3.2.5.1 Database

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.4.1 Database. This document is found in the MEDITI3G Project’s SharePoint site.

3.3.2.5.2 Storage

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.4.2 Storage. This document is found in the MEDITI3G Project's SharePoint site.

3.3.2.5.3 Computing

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.2.4.3 Computing. This document is found in the MEDITI3G Project's SharePoint site.

3.3.3 Network Architecture

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.3 Network Architecture. This document is found in the MEDITI3G Project's SharePoint site.

3.3.3.1 Virtual Network

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.3.1 Virtual Network.

3.3.3.2 VPN Gateway

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.3.2 VPN Gateway. This document is found in the MEDITI3G Project's SharePoint site.

3.3.3.3 Firewall

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.3.3 Firewall. This document is found in the MEDITI3G Project's SharePoint site.

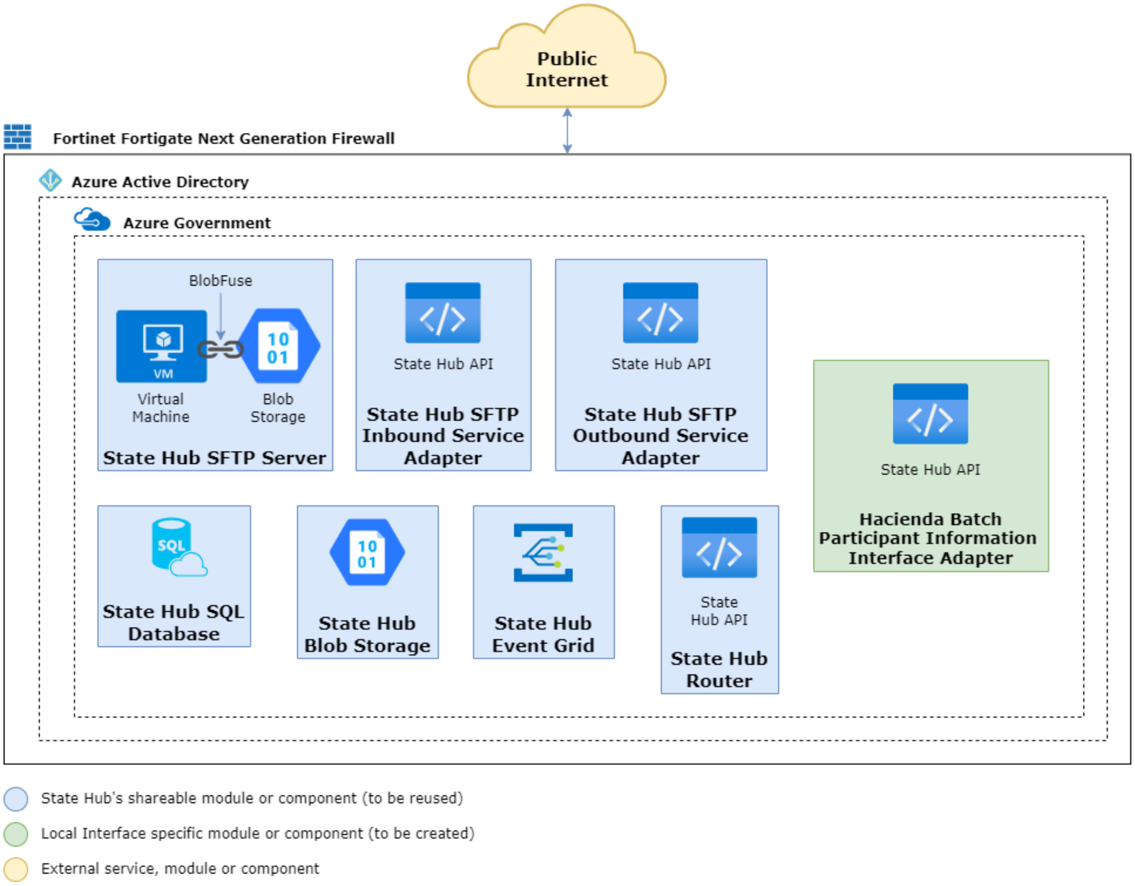
3.3.4 Server Architecture

For details, refer to the document Deliverable 204 - State Hub High-Level Document in the Section 3.3.4 Server Architecture. This document is found in the MEDITI3G Project's SharePoint site.

4 High Level Application Design

The high-level design section provides a more concrete view at how the architecture shall be implemented. It specifies which vendors or services shall be utilized to create the different components of the Local Interface, and how they shall communicate with one to another. Some of these components are part of the State Hub architecture and shall be reused, while others are new and needs development.

Figure 4 - High-Level Application Design


Public Internet
Fortinet Fortigate Next Generation Firewall
Azure Active Directory
Azure Government
BlobFuse
VM
Virtual Machine
10 01
Blob Storage
State Hub SFTP Server
State Hub API
State Hub SFTP Inbound Service Adapter
State Hub API
State Hub SFTP Outbound Service Adapter
State Hub API
Hacienda Batch Participant Information Interface Adapter
SQL
State Hub SQL Database
10 01
State Hub Blob Storage
State Hub Event Grid
State Hub API
State Hub Router
State Hub's shareable module or component (to be reused)
Local Interface specific module or component (to be created)
External service, module or component

4.1 Interface Design

This section provides a more detailed view of how the Local Interface is going to work and the responsibilities of each component, providing sequence diagrams, flow charts, and brief textual explanations to better illustrate the architecture.

Figure 5 - Hacienda Batch Local Interface Sequence Diagram

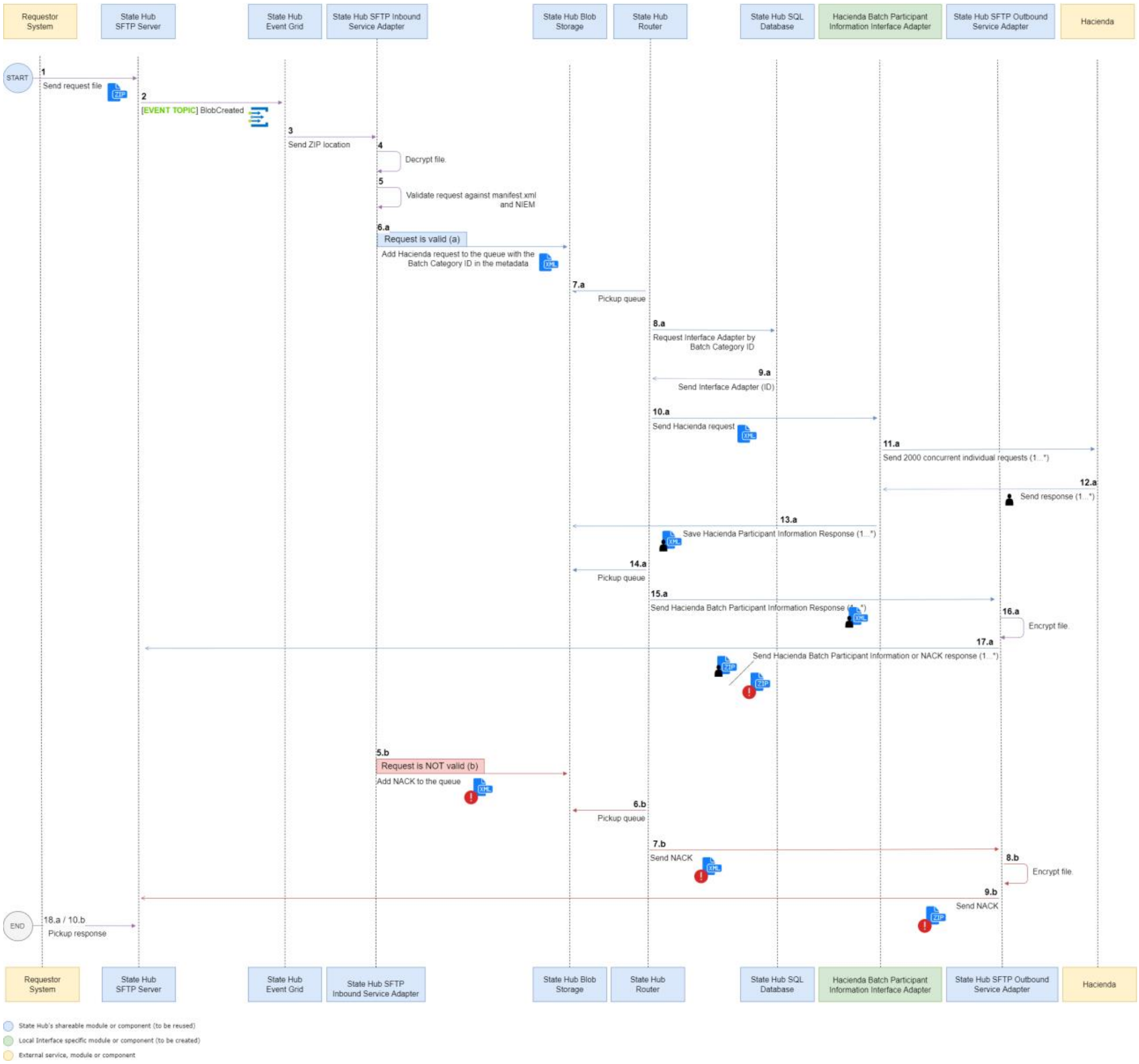

Requestor System
State Hub SFTP Server
State Hub Event Grid
State Hub SFTP Inbound Service Adapter
State Hub Blob Storage
State Hub Router
State Hub SQL Database
Hacienda Batch Participant Information Interface Adapter
State Hub SFTP Outbound Service Adapter
Hacienda
START
1
Send request file
ZIP
2
[EVENT TOPIC] BlobCreated
3
Send ZIP location
4
Decrypt file.
5
Validate request against manifest.xml and NIEM
6.a
Request is valid (a)
Add Hacienda request to the queue with the Batch Category ID in the metadata
XML
7.a
Pickup queue
8.a
Request Interface Adapter by Batch Category ID
9.a
Send Interface Adapter (ID)
10.a
Send Hacienda request
XML
11.a
Send 2000 concurrent individual requests (1...*)
12.a
Send response (1...*)
13.a
Save Hacienda Participant Information Response (1...*)
XML
14.a
Pickup queue
15.a
Send Hacienda Batch Participant Information Response (1...*)
XML
16.a
Encrypt file.
17.a
Send Hacienda Batch Participant Information or NACK response (1...*)
ZIP
ZIP
5.b
Request is NOT valid (b)
Add NACK to the queue
XML
6.b
Pickup queue
7.b
Send NACK
XML
8.b
Encrypt file.
9.b
Send NACK
ZIP
END
18.a / 10.b
Pickup response
Requestor System
State Hub SFTP Server
State Hub Event Grid
State Hub SFTP Inbound Service Adapter
State Hub Blob Storage
State Hub Router
State Hub SQL Database
Hacienda Batch Participant Information Interface Adapter
State Hub SFTP Outbound Service Adapter
Hacienda
State Hub's shareable module or component (to be reused)
Local Interface specific module or component (to be created)
External service, module or component

Figure 6 - Hacienda Batch Local Interface Sequence Diagram Document

PDF
Hacienda Batch
Interface Sequence

Table 3 - Hacienda Batch Local Interface Sequence Table

Step	Message	Description
1	A ZIP file containing the manifest file and the request file	The Requestor System deposits the request ZIP file into the State Hub SFTP Server's Inbound folder
2	A BlobAdded event topic	An event topic is triggered when a file is added to the State Hub SFTP's Inbound folder. The State Hub Event Grid captures the event.
3	A ZIP location	The State Hub Event Grid sends to the State Hub SFTP Inbound Service Adapter the location of the file that was added to the Blob Storage of the State Hub SFTP Server.
4	None	The State Hub SFTP Inbound Service Adapter decrypts the request file.
5	None	The State Hub SFTP Inbound Service Adapter validates that the request file contains all the data that the manifest file says it shall contain. It also validates that the request file satisfies NIEM.
6.a	The XML of the request with the Blob Metadata (see section **Error! Reference source not found.**)	The State Hub SFTP Inbound Service Adapter adds the required metadata to the request XML file and deposits the file into the State Hub Blob Storage.
7.a	An XML file with the requests of the Requestor System and the required metadata	The State Hub Router pickups the XML file and reads the metadata.
8.a	The BatchCategoryCode available as part of the XML file metadata	The State Hub Router uses the BatchCategoryCode to query the State Hub SQL Database to obtain the identifier of the adapter that shall process the file.
9.a	An ID	The State Hub SQL Database returns to the State Hub Router the ID of the adapter that shall process

		the file: the Hacienda Batch Participant Information Interface Adapter.
10.a	An XML file with the requests of the Requestor System and the required metadata	The State Hub Router sends the request XML to the Hacienda Batch Participant Information Interface Adapter.
11.a	None	The Hacienda Batch Participant Information Interface Adapter calls the Hacienda REST Web Service to request the participant information of the individuals requested in the request file. The requests are sent in 2000 concurrent individual requests.
12.a	Participant information data	The Hacienda REST Web Service returns to the Hacienda Batch Participant Information Interface Adapter the participant information of all the requested individuals.
13.a	One or multiple XML files with the participant information data collected from Hacienda and the metadata needed to identify the file as a response and not a request	The Hacienda Batch Participant Information Interface Adapter saves the responses from the REST Web Service Response of the Hacienda Local Agency into different XML files of up to 100 MB of size. Then, the interface adapter saves those files in the State Hub Blob Storage with the metadata needed to identify them as responses.
14.a	None	The State Hub Router pickups the XML files and reads the metadata.
15.a	One or multiple XML files with the participant information data collected from Hacienda and the metadata needed to identify the file as a response and not a request	The State Hub Router routes the XML files to the State Hub SFTP Outbound Service Adapter.
16.a	An encrypted ZIP file with a manifest file and a response file	The State Hub SFTP Outbound Service Adapter validates the response file and creates a new ZIP file with the response and a new manifest file that describes the response. The State Hub SFT Outbound Service Adapter encrypts the generated ZIP with PGP.
17.a	An encrypted ZIP file	The State Hub SFT Outbound Service Adapter deposits the generated ZIP file into the State Hub

		SFTP Server to be picked up by the Requestor System.
5.b	A NACK XML file	The State Hub SFTP Inbound Service Adapter adds the required metadata to the NACK XML file and deposits the file into the State Hub Blob Storage.
6.b	None	The State Hub Router pickups the XML file and reads the metadata.
7.b	A NACK XML file	The State Hub Router sends the response XML to the State Hub SFTP Outbound Service Adapter.
8.b	An encrypted ZIP file with a manifest file and a NACK file	The State Hub SFT Outbound Service Adapter creates a new ZIP file with the response and a new manifest file that describes the NACK. Then, it encrypts the generated ZIP file with PGP.
9.b	An encrypted ZIP file	The State Hub SFT Outbound Service Adapter deposits the generated ZIP file into the State Hub SFTP Server to be picked up by the Requestor System.
18.a /10.b	None	The Requestor System pickups the response file or files from the State Hub SFTP Server.

4.1.1 Hacienda Batch Participant Information Interface Adapter

The Hacienda Batch Participant Information Interface Adapter is responsible for executing the request to the REST Web Service that returns data from the Hacienda Local Agency and returning the data by response files of up to 100 MB of size. The number of individual responses in a 100MB response file shall vary depending on the size of each individual record. It is implemented as part of the State Hub API (C# project).

Figure 7 - Hacienda Batch Participant Information Interface Adapter Module

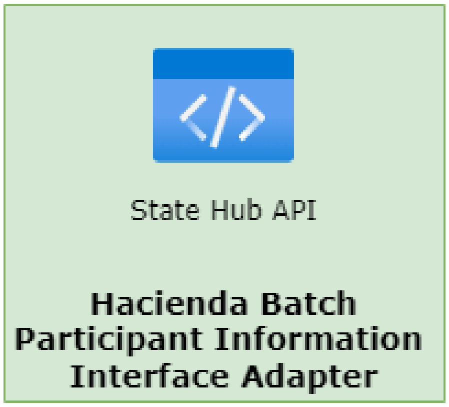
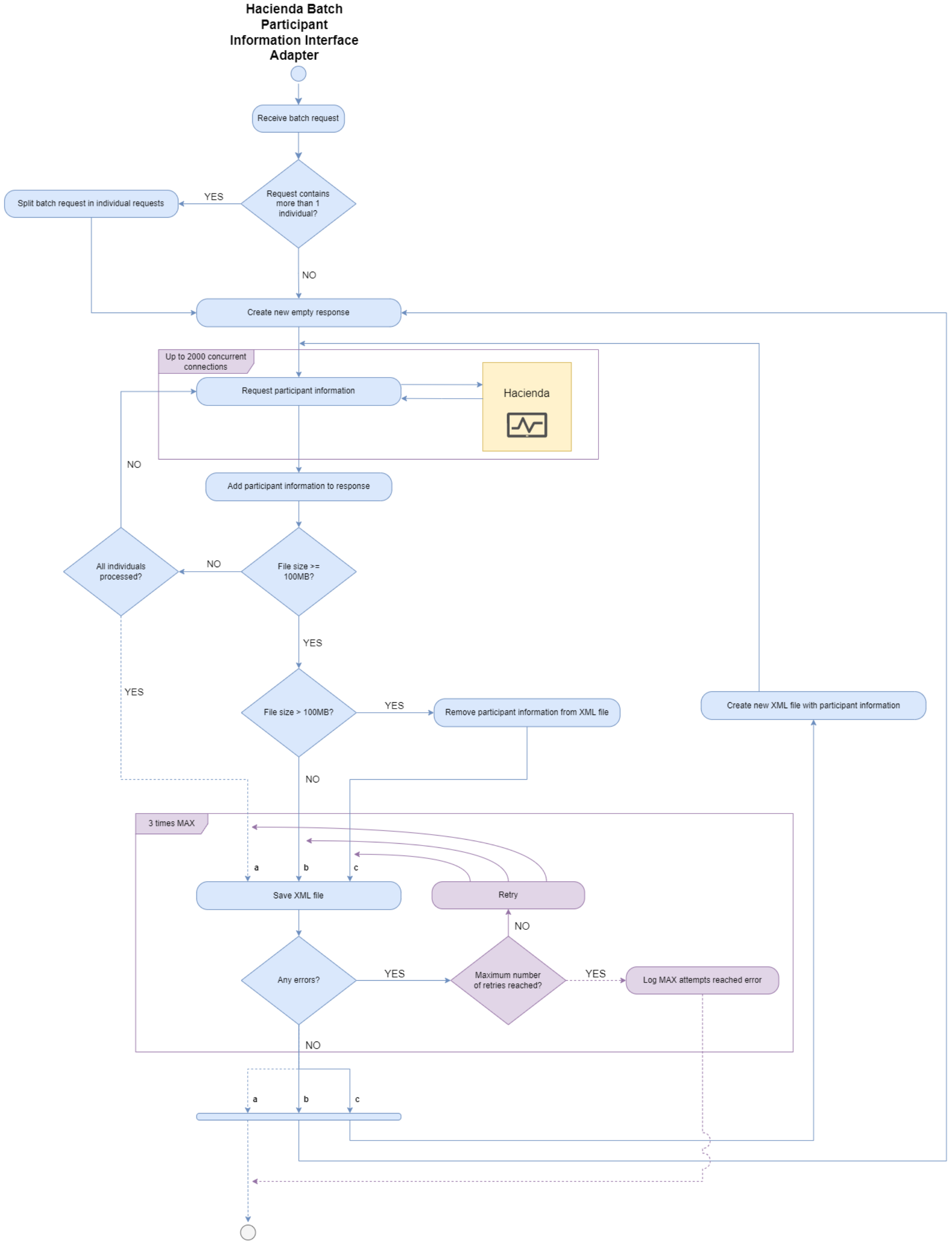


Figure 8 - Hacienda Batch Participant Information Interface Adapter Flow Chart


Hacienda Batch Participant Information Interface Adapter
Receive batch request
Request contains more than 1 individual?
YES
Split batch request in individual requests
NO
Create new empty response
Up to 2000 concurrent connections
Request participant information
Hacienda
NO
Add participant information to response
All individuals processed?
NO
File size >= 100MB?
YES
YES
File size > 100MB?
YES
Remove participant information from XML file
Create new XML file with participant information
NO
3 times MAX
a
b
c
Save XML file
Retry
NO
Any errors?
YES
Maximum number of retries reached?
YES
Log MAX attempts reached error
NO
a
b
c

4.1.2 Hacienda Outbound Participant Information Interface Adapter

The Hacienda Outbound Participant Information Interface Adapter is responsible for decrypting the response with PGP and uploading the Hacienda response file to the State Hub Queue Blob. It is implemented as part of the State Hub API (C# project).

Figure 9 - Hacienda Outbound Participant Information Interface Adapter Module

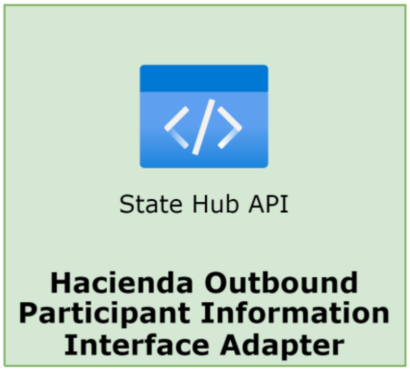
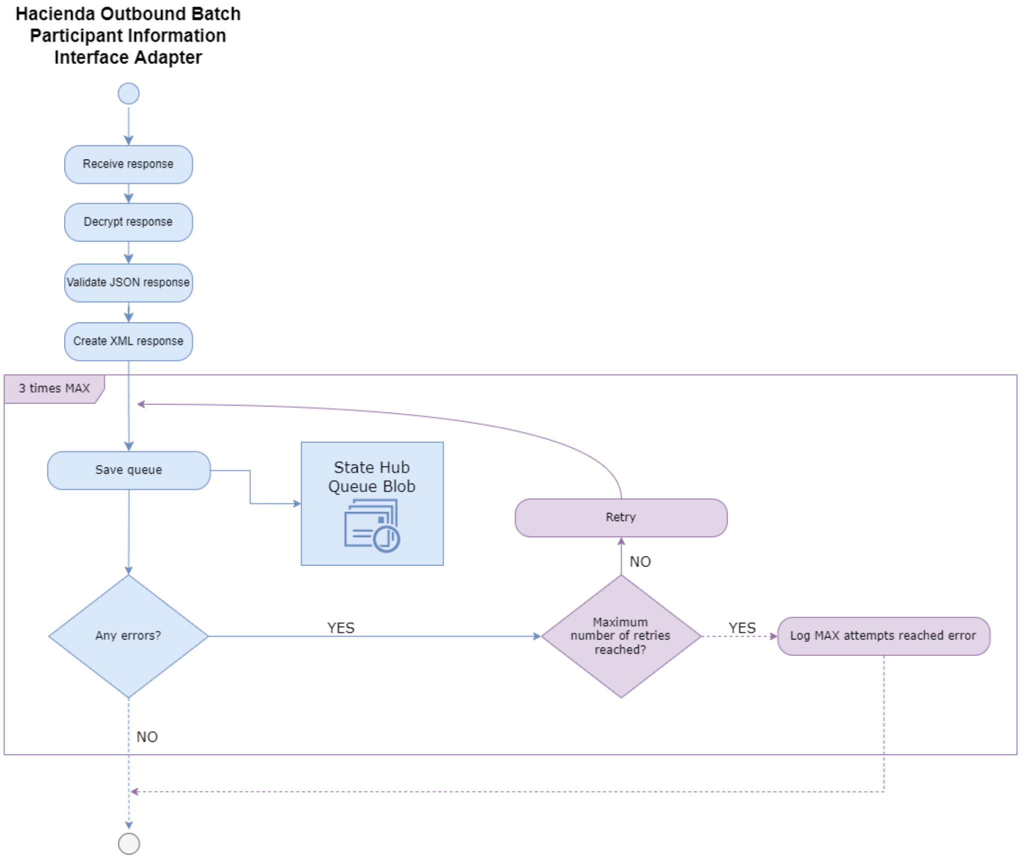


Figure 10 - Hacienda Outbound Participant Information Interface Adapter Flow Chart



4.2 Network Architecture Design

For details on the Network Architecture, refer to the Deliverable 204 - State Hub High-Level Document in section 4.6 *Network Architecture Design*. This section provides information about the Virtual Network, VPN Gateway, Firewall, Application Gateway, and the Domain Controller. This document is found in the MEDITI3G Project's SharePoint.

4.3 Server Architecture Design

For details on the Server Architecture, refer to the Deliverable 204 - State Hub High-Level Document in section 4.7 *Server Architecture Design.* This document is found in the MEDITI3G Project's SharePoint site.

5 Requirements

Requirements for the deliverable in this document are classified, prioritized, and defined in table below.

Table 4 - Requirement Types

ID	Name	Description
AR	Audit requirement	Identify what the State Hub shall keep track of and how to keep track of it, as well as what regulatory or statutory measures are in place to support an audit trail.
F	Functional requirement	Functional requirements define specific functions and/or components that the State Hub shall have. They describe the State Hub behavior.
GR	General requirement	Base requirement that is needed in all scenarios of the design and implementation of the State Hub.
LR	Alert requirement	Identify a behavior specific to how alerts and notifications shall be handled by the State Hub.
MR	Monitoring requirement	Identify how the State Hub shall monitor all activity occurring.
NF	Non-Functional requirement	Non-functional requirements specify criteria that shall be used to judge the operation of the State Hub, rather than specific behaviors.
RR	Reporting requirement	Identify what reports the product and/or system shall be able to manage or what reporting capabilities shall be supported in the State Hub.
SR	Security requirement	Identify the security, confidentiality, integrity, and privacy issues affecting access to the State Hub, and protection of the data that the State Hub creates or uses.
HA	Hacienda requirement	Specific requirement targeted for Hacienda

5.1 Functional Requirements

Functional requirements are features or functions that shall be implemented to enable users to accomplish their tasks or, in other words, what the system shall do. As such, using the above table, the State Hub shall adhere to the following functional requirements:

Table 5 - Functional Requirements

#	ID	Requirement	Note
1.	IR-GR-F-001	The real-time and batch local interfaces shall be hosted in the cloud as services.	
2.	IR-GR-F-003	The batch local interfaces shall receive batch requests from the MEDITI 3G System and route them to their corresponding local agency.	
3.	IR-GR-F-005	The batch local interfaces shall deliver batch responses from their corresponding local agency to the MEDITI 3G System.	
4.	IR-GR-F-008	The batch local interfaces shall be able to accept one or more batch files for processing.	
5.	IR-GR-F-009	The batch local interfaces shall send NACKS when a request was not processed due to a validation error, whether NIEM validation error, checksum validation error, or otherwise.	
6.	IR-GR-F-010	The batch local interface shall return a response if the max agreed upon response time is exceeded. The response shall either be a NACK if whole response file has yet to be processed, or a response batch file with individual responses based on batch processing completed against agency at that time.	
7.	IR-GR-F-011	The real-time and batch local interfaces shall be uniquely	

		identifiable from within the State Hub such that audit trails, log files, reporting services and other transactions shall be quickly identified per local interface by the administrator user and auditor user when performing administrative tasks from the cloud portal.	
8.	IR-GR-F-012	The real-time and batch local interfaces shall process all PII in transit and shall not retain any PII after the processing is completed.	
9.	IR-GR-F-013	The batch local interfaces shall allow partial responses for batch transactions.	All responses shall be identified by their Request ID that is generated in the MEDIT3G system to correlate each request with its appropriate response.
10.	IR-GR-F-015	The real-time and batch local interfaces shall support transporting inbound and outbound data to the MEDITI 3G system adhering to the NIEM 2.0 standard.	
11.	IR-GR-F-016	The real-time and batch local interfaces shall send PII as search criteria to locate the person/participant at the local agency.	
12.	IR-GR-F-017	The real-time and batch local interfaces shall receive a response from their respective local agency with the participant information pre-defined data elements.	
13.	IR-GR-F-018	The batch local interface shall allow MEDITI 3G System to submit a batch request file for querying the local agency System for Participant(s) Information, to be returned within one or more response files.	
14.	IR-GR-F-020	The batch local interface request file shall contain a batch set of individual participant requests, each request	

		containing participants PII search criteria.	
15.	IR-AR-F-002	The batch local interfaces shall log events resulting from requests received from the MEDITI 3G System through the State Hub and the response from their corresponding local agency. At a minimum, events that shall be logged are: 1. Batch file received for batch transactions. 2. Size of the batch ZIP file in Kilo Bytes (KB), Mega Bytes (MB), or Giga Bytes (GB) 3. Size of the batch file's XML document in KB, MB, or GB 4. File validation results. a. Requester ID captured 5. Request transformation results (optional). 6. Result of connectivity attempt to the local agency. a. Connection was established to the local agency (timestamp). b. Connection timeout between interface and local agency. 7. Agency query results a. Error code 8. Response transformation results (optional) 9. Transaction completed after transmitting data to the local agency. a. Correlation ID captured. 10. File placed for pick-up. 11. File picked-up. File auto-removed.	
16.	IR-AR-F-003	The real-time and batch local interfaces shall log error codes accompanied by an unvarying, standard description that defines what the error code means when an exception occurs.	
17.	IR-AR-F-004	The real-time and batch local interfaces shall relay audit trails related to warnings and errors to the State Hub using a normalized coding structure so that they are easily	

		identifiable for auditing and troubleshooting purposes.	
18.	IR-AR-F-005	The real-time and batch local interfaces shall not store PII in audit trails.	
19.	IR-AR-F-006	The real-time and batch local interfaces shall capture non-personal identifying invalid data in the communication (request and response) to help with troubleshooting.	
20.	IR-SR-F-001	The real-time and batch local interfaces shall ensure that if a failure occurs, no sensitive information, such as PII, vulnerable to external attacks via interface responses or captured audit trail.	
21.	IR-SR-F-002	The real-time and batch local interfaces shall keep data encrypted during transit as originated from the MEDITI 3G System and the Local Agency.	
22.	IR-SR-F-003	The real-time and batch local interfaces shall establish a secure connection with the MEDITI 3G System and the Local Agency.	
23.	IR-SR-F-004	The batch local interfaces shall keep data encrypted at rest while the transaction is being processed.	
24.	IR-SR-F-005	The batch local interfaces shall permanently remove all batch files, after the request has been processed and the response has been sent to the MEDITI 3G System.	
25.	IR-SR-F-006	The real-time and batch local interfaces shall comply with the security guidelines and recommendations established in the Patient Protection and Affordable Care Act of 2010, Section 1561.	
26.	IR-SR-F-007	The real-time and batch local interfaces shall comply with the	

		security requirements established by the HITECH 2009.	
27.	IR-SR-F-008	The real-time and batch local interfaces shall restrict access to appropriately authenticated systems (for example, MEDITI 3G System and Local Agencies' Systems).	
28.	IR-SR-F-009	The real-time and batch local interfaces shall restrict access to appropriately authenticated users (for example, administrator and auditor).	
29.	IR-SR-F-010	The real-time and batch local interfaces shall allow an administrator, without granting read access, to delete an in-transit file (stuck in-transit).	
30.	IR-SR-F-011	The batch local interfaces shall securely purge (delete) any file that reaches or surpasses the predefined time for processing.	
31.	IR-GR-F-HA-001	The Hacienda real-time and batch local interfaces shall support the ability to retry a transaction, without manual intervention, after the local agency becomes unavailable mid-transaction.	
32.	IR-GR-F-HA-002	The Hacienda real-time and batch local interfaces shall capture metric of whether the local agency endpoint is online or unavailable at the time of its use, up to including any retry attempts.	
33.	IR-GR-F-HA-004	The HACIENDA batch local interface shall validate batch files submitted by MEDITI 3G System for message format compliance and integrity.	
34.	IR-GR-F-HA-005	In case of connectivity issues between the batch local interface and the local agency, the local interface shall retry establishing connection and processing the transaction every half-hour (30 minutes) for up to the	

		max response time or a consecutive 24-hour window of not communicating. Each attempt of reconnecting shall be notified to the State Hub.	
35.	IR-GR-F-HA-006	HACIENDA real-time and batch local interfaces shall provide the data elements in section 11.1 upon request, whether by real-time or batch.	

5.2 Non-functional Requirements

Non-functional requirements describe how a system shall behave and establish constraints of its functionality. As such, the State Hub shall adhere to the following non-functional requirements:

Table 6 - Non-Functional Requirements

#	ID	Requirement
1.	IR-GR-NF-004	The batch local interfaces shall process batch uncompressed XML files that do not exceed two hundred fifty (250) Megabytes (MB).
2.	IR-GR-NF-005	The batch local interfaces shall be able to process up to one (1) concurrent batch request file.
3.	IR-GR-NF-006	The batch local interface shall expose an SFTP directory so that MEDITI 3G System shall submit batch requests files for batch querying.
4.	IR-GR-NF-007	The batch local interface shall expose an SFTP directory so that MEDITI 3G System shall pick up any batch response files destined for MEDITI 3G System.
5.	IR-GR-NF-008	The batch local interfaces shall permanently remove in-transit files that has not been used within ten (10) calendar days.
6.	IR-GR-NF-009	The real-time and batch local interfaces shall comply with HIPAA and MARS-E regulations to guarantee data encryption, protection, portability, and integrity.
7.	IR-GR-NF-012	The batch interfaces shall support Application-to-Application asynchronous behavior for batch requests.

8.	IR-LR-NF-001	The real-time and batch local interfaces shall generate alerts and notifications through the State Hub using monitoring capabilities.
9.	IR-MR-NF-001	The real-time and batch local interfaces shall capture metrics on the availability of the service provider (local agency). The metric shall compliment the State Hub's service provide monitoring capabilities.
10.	IR-SR-NF-001	The real-time and batch local interfaces that support Secure Socket Layer (SSL) connections shall be supported by public key/private key encryption SSL certificates with 256-bit encryption or stronger.
11.	IR-SR-NF-002	The security configurations and conditions that the real-time and batch local interfaces are required to implement in a production environment shall be the same configurations and conditions implemented in all development, testing, integration, and acceptance test environments to guarantee compliance with the security measures in the MARS-E for protecting PII.
12.	IR-SR-NF-003	The real-time and batch local interfaces development and development tests shall not use real data for development or testing environments.
13.	IR-SR-NF-004	The batch local interfaces shall perform source to destination file integrity checks for exchange of data to ensure no corrupted data reaches to or is extracted from the local agency.
14.	IR-GR-NF-HA-001	The batch local interface shall connect to Hacienda's batch endpoint using their existing SFTP environment.
15.	IR-GR-NF-HA-002	The real-time and batch local interfaces shall receive a response from HACIENDA with the following participant information data elements: a. Income snapshot b. Resources (Non-MAGI) c. Address Identification Details
16.	IR-GR-NF-HA-003	The HACIENDA batch local interfaces shall have a maximum response time of five (5) days.
17.	IR-GR-NF-HA-004	In case of connectivity issues between the batch local interface and the local agency, the local interface shall retry establishing connection and processing the transaction every half-hour (30 minutes) for up to the max response time or a consecutive 24-hour window of not communicating. Each attempt of reconnecting shall be notified to the State Hub.

18.	IR-SR-NF-HA-001	The HACIENDA real-time and batch local interfaces shall be hosted in a MARS-E secure environment.

5.3 MARS-E Control Standards 2019

The embedded document below, the **MARS-E Control Standards 2019-Wovenware.xlxs**, is a spreadsheet that indicates which of the controls specified in the CMS MARS-E apply directly to the State Hub's design and implementation as described in this document, which makes for a total of 29 controls. The spreadsheet contains several tabs, each identified with the specific control it represents and what controls were identified by the PMO Security Director for Wovenware to comply with.

Table 7 – MARS-E Control Standards Summary

Control Domain	**Total Controls for MARS-E for this deliverable**	**WW Requirements**
Access Controls (AC)	7	7
Audit and Accountability (AU)	3	1
Security Assessment and Authorization (CA)	1	1
Configuration Management (CM)	1	1
Contingency Planning (CP)	1	0
Identification and Authentication (IA)	2	2
Incident Response (IR)	2	1
Media Protection (MP)	3	3
Physical and Environmental Protection (PE)	2	2
Planning (PL)	1	1
Personnel Security (PS)	2	2
System and Service Acquisition (SA)	2	2
System and Communications Protection (SC)	3	3
System and Information Integrity (SI)	2	2
Minimization of PII Used in Testing, Training, and Research/Risk Minimization Techniques (Enhancement) (DM-3 (1)) –PRIVACY CONTROL	15	1
Totals	**47**	**29**

The attached file below contains all the CMS controls under MARS-E that Wovenware shall comply with.

MARS-E Control
Standards 2019-Wov