

# **Beneficiary Travel Self-Service System (BTSSS)**

## **System Design Document**



**November 2015**

**Version 1.0**

**Department of Veterans Affairs**

## Revision History

Date	Version	Description	Author
11-18-15	1.0	Initial Draft	
11-13-15	0.12	Add sections 4.3,4.4 and 4.5	
11-13-15	0.11	Added Section 4.2	
11-13-15	0.10	Edited Section 2	
11-09-15	0.9	Added TO-BE Diagram	
11-09-15	0.8	Added Section 2	
11-06-15	0.7	Updated Appendices 1 & 2	
11-06-15	0.6	Inserted network diagram to Section 6.3	
11-05-15	0.5	Updated Sections 6.4.1 & 6.4.2	
11-04-15	0.4	Updated Sections 1.1 & 1.2	
11-04-15	0.3	Updated Sections 5 & 6-6.2.2.3.33	
11-03-15	0.2	Updated Introduction, Appendices 1 & 2	
10-30-15	0.1	Updated Section 3	

## Artifact Rationale

The System Design Document (SDD) is a dual-use document that provides the conceptual design, as well as the as-built design. This document will be updated as needed, to reflect the as-built product.

## When to Complete Each Section of the SDD

Section	Completed On or Before PMAS Phase	Rationale
1 – Introduction	MS 0 Review; updated thereafter	Conceptual design should inform evaluation of investments
2 - Background	MS 0 Review; updated thereafter	Conceptual design should inform evaluation of investments
3 – Conceptual Design	MS 0 Review; updated thereafter	Conceptual design should inform evaluation of investments
4 – System Architecture	MS 0 Review; updated thereafter	Conceptual design should inform evaluation of investments
5 – Data Design	MS 1 Review; updated thereafter	Design details should be elaborated upon during PMAS Planning phase and prior to development
6 – Detailed Design	MS 1 Review; updated thereafter	Design details should be elaborated upon during PMAS Planning phase and prior to development
7 – External System Interface Design	MS 1 Review; updated thereafter	Design details should be elaborated upon during PMAS Planning phase and prior to development
8 – Human Machine Interfaces	MS 1 Review; updated thereafter	Design details should be elaborated upon during PMAS Planning phase and prior to development
Attachments	MS 1 Review; updated thereafter	Design details should be elaborated upon during PMAS Planning phase and prior to development

A product's system design should be defined conceptually prior to the allocation of personnel and resources that occur at project initiation. This gives the enterprise an opportunity to evaluate IT investments before project teams are stood up and funding is allocated. Sections 1-4, which discuss the high level design, should be completed prior to MS 0. All sections should be completed and updated before MS 1. Projects will need to address all SDD approval constraints prior to the MS 2 review. In addition, the SDD should reflect the as-built product going into the MS 2 review.

## Instructions

Activity	New Capability (1)	Feature Enhancement (2)
Field Deployment (A)	Yes	Yes
Cloud/Web Deployment (B)	Yes	Yes
Mobile Application (C)	Yes	Yes



## Table of Contents

<b>1. Introduction .....</b>	<b>1</b>
1.1. Scope.....	1
1.2. User Profiles.....	3
<b>2. Background .....</b>	<b>4</b>
2.1. Overview of the System .....	5
2.2. Overview of the Significant Requirements .....	6
2.2.1. Overview of Significant Functional Requirements .....	9
2.2.2. Overview of Functional Workload/Performance Requirements.....	11
2.2.3. Overview of Operational Requirements .....	11
2.2.4. Overview of Pivotal Technical Requirements .....	12
2.2.5. Overview of the Security and Privacy Requirements .....	12
2.2.6. Overview of System Criticality and High Availability Requirements.	13
2.2.7. Overview of Single Sign on Requirements .....	13
2.2.8. Overview of Enterprise Portals Requirements .....	13
2.2.9. Overview of Special Device Requirements.....	13
<b>3. Conceptual Design.....</b>	<b>14</b>
3.1. Conceptual Application Design .....	14
3.1.1. Application Context .....	14
3.1.2. High-Level Application Design .....	19
3.1.3. Application Locations .....	21
3.2. Conceptual Data Design.....	21
3.2.1. Project Conceptual Data Model .....	21
3.2.2. Database Information .....	22
3.2.3. User Interface Data Mapping .....	22
3.2.3.1. Application Screen Interface .....	22
3.2.3.1.1. <Insert name of screen> .....	22
3.2.3.2. Application Report Interface.....	23
3.2.3.2.1. <Insert name of report>.....	23
3.2.3.3. Unmapped Data Element.....	24
3.3. Conceptual Infrastructure Design .....	24
3.3.1. System Criticality and High Availability.....	24

3.3.2.	Special Technology .....	25
3.3.3.	Technology Locations.....	25
3.3.4.	Conceptual Infrastructure Diagram.....	26
3.3.4.1.	Location of Environments and External Interfaces .....	26
3.3.4.2.	Conceptual Production String Diagram.....	26
<b>4.</b>	<b>System Architecture .....</b>	<b>27</b>
4.1.	Hardware Architecture .....	27
4.2.	Software Architecture.....	28
4.3.	Network Architecture.....	32
4.4.	Service Oriented Architecture / ESS .....	32
4.5.	Enterprise Architecture .....	32
<b>5.</b>	<b>Data Design.....</b>	<b>33</b>
5.1.	DBMS Files .....	33
5.2.	Non-DBMS Files .....	33
5.3.	Data View.....	34
<b>6.</b>	<b>Detailed Design .....</b>	<b>34</b>
6.1.	Hardware Detailed Design.....	34
6.2.	Software Detailed Design.....	34
6.2.1.	Conceptual Design .....	34
6.2.1.1.	Product Perspective .....	35
6.2.1.1.1.	User Interfaces .....	35
6.2.1.1.2.	Hardware Interfaces .....	35
6.2.1.1.3.	Software Interfaces .....	35
6.2.1.1.4.	Communications Interfaces .....	35
6.2.1.1.5.	Memory Constraints.....	36
6.2.1.1.6.	Special Operations .....	36
6.2.1.2.	Product Features.....	36
6.2.1.3.	User Characteristics.....	36
6.2.1.4.	Dependencies and Constraints .....	36
6.2.2.	Specific Requirements .....	37
6.2.2.1.	Database Repository.....	37
6.2.2.2.	System Features .....	38

6.2.2.3. Design Element Tables .....	38
6.2.2.3.1. Routines (Entry Points) .....	38
6.2.2.3.2. Templates .....	40
6.2.2.3.3. Bulletins .....	41
6.2.2.3.4. Data Entries Affected by the Design.....	42
6.2.2.3.5. Unique Record(s) .....	42
6.2.2.3.6. File or Global Size Changes .....	43
6.2.2.3.7. Mail Groups .....	43
6.2.2.3.8. Security Keys .....	44
6.2.2.3.9. Options .....	46
6.2.2.3.10. Protocols .....	48
6.2.2.3.11. Remote Procedure Call (RPC) .....	49
6.2.2.3.12. Constants Defined in Interface .....	50
6.2.2.3.13. Variables Defined in Interface .....	50
6.2.2.3.14. Types Defined in Interface .....	50
6.2.2.3.15. GUI.....	50
6.2.2.3.16. GUI Classes .....	51
6.2.2.3.17. Current Form .....	51
6.2.2.3.18. Modified Form .....	51
6.2.2.3.19. Components on Form .....	51
6.2.2.3.20. Events .....	51
6.2.2.3.21. Methods .....	51
6.2.2.3.22. Special References .....	52
6.2.2.3.23. Class Events.....	52
6.2.2.3.24. Class Methods.....	52
6.2.2.3.25. Class Properties.....	52
6.2.2.3.26. Uses Clause.....	52
6.2.2.3.27. Forms.....	52
6.2.2.3.28. Functions .....	53
6.2.2.3.29. Dialog .....	55
6.2.2.3.30. Help Frame .....	56
6.2.2.3.31. HL7 Application Parameter.....	56
6.2.2.3.32. HL7 Logical Link .....	57

6.2.2.3.33. COTS Interface .....	58
<b>6.3. Network Detailed Design.....</b>	<b>59</b>
<b>6.4. Security and Privacy.....</b>	<b>59</b>
6.4.1. Security.....	59
6.4.2. Privacy .....	59
<b>6.5. Service Oriented Architecture / ESS Detailed Design .....</b>	<b>60</b>
6.5.1. Service Description for <Consumed Service Name>.....	60
6.5.2. Service Design for <Provided Service Name> .....	60
6.5.2.1. Introduction .....	60
6.5.2.1.1. Purpose and Scope of Service .....	60
6.5.2.1.2. Links to Other Documents .....	61
6.5.2.2. Service Details .....	61
6.5.2.2.1. Service Identification .....	61
6.5.2.2.2. Service Versions .....	62
6.5.2.2.3. Summary of Design and Platform Details.....	62
6.5.2.2.3.1. SOA Pattern(s) Implemented .....	62
6.5.2.2.3.2. COTS Platform Vendor Names and Versions for Hosting Platform ....	62
6.5.2.3. Dependencies .....	63
6.5.2.4. Service Design Details .....	63
6.5.2.4.1. Interface Technical Specs.....	63
6.5.2.4.1.1. Service Invocation Type .....	63
6.5.2.4.1.2. Service Interface Type .....	63
6.5.2.4.1.3. Service Name .....	63
6.5.2.4.1.4. Interface .....	63
6.5.2.4.1.5. End Points .....	63
6.5.2.4.1.6. Operations or Methods.....	63
6.5.2.4.1.7. Message Schemas .....	64
6.5.2.4.2. Information Model .....	64
6.5.2.4.2.1. Class Diagram and Description of Entities Involved .....	64
6.5.2.4.2.2. Mappings from ELDM to Standards Based Schemas .....	64
6.5.2.4.3. Behavior Model (AKA Use Case Realization).....	64
6.5.2.4.3.1. Use Cases (Use Case Model) .....	64
6.5.2.4.3.2. Interaction Diagrams .....	64

6.5.2.5. Gap Analysis.....	65
6.5.2.5.1. Variances from Enterprise Target Architecture.....	65
6.5.2.5.2. Variances from SLDs.....	65
6.5.2.5.3. Variances from Standards and Policies.....	65
6.5.2.5.4. Justification for Exceptions and Mitigation .....	65
<b>7. External System Interface Design .....</b>	<b>66</b>
7.1. Interface Architecture.....	66
7.2. Interface Detailed Design .....	66
<b>8. Human-Machine Interface .....</b>	<b>68</b>
8.1. Interface Design Rules .....	68
8.2. Inputs .....	68
8.3. Outputs .....	68
8.4. Navigation Hierarchy .....	68
8.4.1. Screen [x.1] .....	68
8.4.2. Screen [x.2] .....	68
8.4.3. Screen [x.3] .....	68
<b>9. Attachment A – Approval Signatures .....</b>	<b>69</b>
<b>A. Additional Information .....</b>	<b>70</b>
A.1. Identification of Technology and Standards .....	70
A.2. Constraining Policies, Directives and Procedures .....	70
A.3. Requirements Traceability Matrix.....	70
A.4. Packaging and Installation.....	80
A.5. Design Metrics .....	80

# 1. Introduction

The Department of Veterans Affairs (VA) operates the nation's largest integrated health care network, operating over 150 hospitals, over 130 community living centers, and over 900 outpatient clinics. There are over 8.5 million Veterans enrolled in the VA health care system, with over 5.5 million Veterans using the Veterans Health Administration (VHA) health care system in any given year. In addition to the health care facilities available for the Veteran, the VA has authority to provide, to eligible persons, reimbursement for mileage driven in a private vehicle and transportation by common carrier/public transportation (plane, bus, taxi, etc.). In addition, when medically justified by a VA health care provider, special mode of transportation (ambulance, wheelchair van, etc.) may be approved for Beneficiary Travel (BT)-eligible Veterans. The eligible Beneficiary requests travel reimbursement through the BT program.

In 1978, the VA travel mileage reimbursement rate was 11 cents per mile; the rate was increased to 28.5 cents per mile in February 2008; and increased again, in November 2008, to 41.5 cents per mile. As a result, BT experienced significant growth in both use and cost. BT is now one of the most highly funded federal transportation programs for people with special needs. In December 2009, the VHA reported that, following the mileage rate reimbursement increase in November 2008, mileage claims grew by 76 percent, with a 30 percent increase in the number of Veterans claiming travel reimbursements.

The VHA estimates that if all Veterans eligible for mileage reimbursement sought it, the VA would incur expenses in excess of \$1.5 billion per year. BT obligations were approximately \$861 million in FY 2012; starting in 2010, the VHA began a series of initiatives to improve oversight of the BT and travel reimbursement claims processing. The Beneficiary Travel Self-Service System (BTSSS) is a portion of the BT overall improvement effort.

The VA seeks to advance the BT program by creation of a web-based beneficiary self-service application that will automate claims submission and travel reimbursement payment. The current travel reimbursement process is manual, time consuming, and requires the assistance of a VA clerk. The BTSSS will continue the efforts of the VA to improve its services to Veterans, and reduce long-term costs, by streamlining the travel claim and reimbursement process.

The BTSSS will provide features and capabilities that leverage automation and multiple-user interface capabilities to manage and process Beneficiary Travel claims, common in commercial software. The VA plans to acquire a commercial-off-the-shelf (COTS) software product that can be customized to integrate with the VA's Veteran identification systems for records and the appointment scheduling systems.

## 1.1. Scope

The scope of the BTSSS is to implement a solution that replaces the existing BT functions. The foundation of the BTSSS is COTS software, customized for processing Veteran and Beneficiary reimbursement claims for mileage, transportation, meals, and lodging expenses. The BTSSS adds the following key features and capabilities:

**Table 1: Scope Inclusions**

<b>Key Capability</b>	<b>Description</b>
Support for Electronic Funds Transfer payments	A central Business need is to eliminate BT cash reimbursement payments and to alleviate the lines at the facilities. The BTSSS will be a key to moving to this new cashless model.
Support for Veteran and Non-Veteran Beneficiary Travel Claimants	The BTSSS will support the creation of BT profiles for both Veterans and Non-Veteran BT Claimants to ensure efficient processing and payment.
Support for Multiple User Interfaces and Modes	The BTSSS will be accessible via the following user interfaces: <ul style="list-style-type: none"><li>– A standard desktop web browser (for example, PC Desktop, laptop, etc.). This is to ensure quick deployment of the solution and immediate use;</li><li>– A user Interface optimized for mobile devices (for example, mobile browser);</li><li>– A user Interface optimized for use on Kiosk devices.</li></ul>
Support for Submission of Receipts and Non-Receipt Expenses	The BTSSS will support the ability to submit expenses that require a receipt (for example, Air Travel) and expenses that do not require are receipt (for example, Mileage expense).
Support for Submission of Expense Report	The BTSSS will support the ability to submit an expense report composed of expense entries with scanned images of the supporting documentation. The supporting documentation is used to document expenses for Meals/Lodging, Other Expenses (for example, Tolls), other Transportation modes (for example, Air Travel), and Special Mode.
Ability to plan for future Beneficiary Travel	<p>The BTSSS will need to provide the BT Claimant the ability to plan for future trips. This would entail allowing the BT Claimant to submit information for a particular future appointment and receive information from the BTSSS on how the claim may be adjudicated (for example, the eligibility of the trip for BT reimbursement and, if eligible, the estimated reimbursement amount).</p> <p>In addition, the BTSSS will allow the BT Claimant to create a claim for a future appointment and submit it for payment. However, the BTSSS will not process or calculate the payment until the appointment has been completed. The purpose of this feature is to reduce the numbers of steps between planning to travel and submitting a claim.</p>
Business Intelligence and Reporting	The BTSSS will need to include the ability to produce a range of reports to ensure that the solution is effective and meets the goals and objectives of the Business Owners.

## 1.2. User Profiles

The table contains the descriptions and characteristics of the intended users of the BTSSS:

**Table 2: User Roles and Access Levels**

User Level	Role	Responsibilities	BTSSS Capabilities Access Level
Primary	Beneficiary Travel Claimants	Represents a Veteran, Caregiver, or other party that is requesting Beneficiary Travel reimbursement.	<ul style="list-style-type: none"><li>– Enter/View/Edit profile;</li><li>– Enter/View/Edit claims (reimbursement requests)</li></ul>
Primary	Travel Clerk	Represents the person(s) responsible for assisting the Beneficiary Travel Claimant with issues with their claim.  May need to process the claim manually in some cases when exceptions or special situations are encountered.	<ul style="list-style-type: none"><li>– Enter/View/Edit profile;</li><li>– Enter/View/Edit claims (reimbursement requests);</li><li>– Approve requests;</li><li>– Trigger/Send notifications.</li></ul>
Primary	Business User	Represents the person who utilizes the BTSSS data for business intelligence and reporting analysis.	<ul style="list-style-type: none"><li>– Run/View reports</li></ul>
Secondary	Application Super User	Represents the person who is engaged in report design/customization, workflow design/modification, and parameters configuration setup/modification.	<ul style="list-style-type: none"><li>– Design/Edit Reports;</li><li>– Design/Edit Workflow;</li><li>– Enter/Edit Configurable parameters.</li></ul>
Secondary	System Administrator	Represents the person who has full control on the system.	Full control.



## 2. Background

Not all Veterans are eligible for travel benefits, and not all travel costs are covered by the BT program. The VA determination of eligibility for BT program benefits are based on the characteristics of the Veteran, generally related to service connection and income levels, and the type of medical appointment, or a combination of the criteria. Time limits also apply for requesting reimbursement. A limited group of non-Veterans is also eligible for reimbursement of some travel costs related to medical appointments at VA facilities.

Travel benefits are provided only for care that is being paid for by the VA, and only for care that has been previously scheduled, unless it is emergency care. Those eligible for benefits through the program receive a per-mile payment for travel by car, reimbursement for “special mode” transportation when justified, and in some circumstances reimbursement for air travel. Veterans must submit requests for reimbursement within 30 days of the travel, although certain cost reimbursements must be requested before the travel takes place. Deductibles apply in most cases.

The current BT process requires claimants to apply in person or in writing within 30 calendar days of completing the travel. Veterans and other beneficiaries may apply for travel reimbursement at the VA medical center in which their appointment was held, through a travel office or business office within the facility. Medical centers also establish procedures to provide benefits for eligible Veterans traveling to Community Based Outpatient Clinics (CBOC) or other facilities under their jurisdiction. The reimbursement request uses Veteran information stored in the Veterans Health Information Systems and Technology Architecture (VistA), and includes the Veteran’s home address, and verification that the Veteran had a qualifying appointment on the request date for travel reimbursement.

VA clerks determine mileage to be reimbursed, which can be one-way or round-trip. The mileage is entered into the VistA Beneficiary Travel Package, which computes the amount payable. The beneficiary signs the payment voucher certifying the claim is correct. Beneficiaries present the form to the agent cashier for cash reimbursement. Other methods of payment include by check and Electronic Funds Transfer.

The VHA recognizes the need to improve financial management for BT payments to reduce the risk of fraudulent payments. As examples, the VHA did not perform regular reconciliations of approved travel reimbursements with paid reimbursements, nor accurately code financial transactions. This occurred because VHA had not established policies and mechanisms that addressed reconciliations of BT financial data, provided adequate training to ensure accurate coding of beneficiary travel expenses, nor established procedures to mitigate the risk for making duplicate payments on approved travel reimbursements. In addition, information system limitations present challenges to performing automated reconciliations.

As a result of inadequate financial controls, VHA lacks assurance that its liabilities, expenditures, and the full costs of BT are consistently and accurately recorded, monitored, and uniformly reported. An audit by the Office of the Inspector General (OIG), in February 2013, recommended the Under Secretary for Health strengthen authorization, payment, and oversight controls for the Beneficiary Travel Program.

## 2.1. Overview of the System

The new travel reimbursement solution, the BTSSS, will interface with existing VA systems that provide Veteran information, identity and access control, and electronic funds transfer (EFT) information. The Veteran, or a non-Veteran claimant, will be able to submit their own claims, receive claim status, and file an appeal. The VHA will be able to generate liability, expenditure, and other BT reports to provide the oversight mandated by OIG. The BTSSS needs to be web-based and accessible through the following means:

- Using standard web-browsers;
- Being integrated with VA self-service portals, such as MyHealtheVet and eBenefits;
- Using mobile devices; and
- Using Veteran Point of Service (VSP) Kiosks located in VA medical facilities.

The BTSSS will need to provide many of the features and capabilities found in the typical third-party expense management services and solutions. These can be summarized as:

- The ability to create a user profile with contact and financial information, which is used for EFT payment.
- The ability to create expense entries.
- The ability to upload scanned images of receipts and supporting documentation and links these to the expense entries.
- The ability to plan for future Beneficiary Travel.
- The ability to submit expenses for either manual or automatic review for approval.
- The ability to receive the payment in the form of an EFT transaction.
- The ability to improve Business Intelligence and Reporting.

The main difference between the BTSSS and the typical third-party expense management services and solutions is the need to evaluate claims based on the VA Beneficiary Travel rules and guidelines, and using Medical Appointment and Eligibility information found in external VA systems.

The [BTSSS Requirements Specification Document \(RSD\)](#) specifies the functional and non-functional requirements of the BTSSS. The BTSSS solution replaces the existing BT program functions performed by the current process and systems, while adding the following key features and capabilities:

- **Support for EFT Payments**

A central Business need is to eliminate BT cash reimbursement payments and to alleviate the lines at the facilities. The BTSSS will be a key to moving to this new cashless model.

- **Support for Veteran and Non-Veteran BT Claimants**

The BTSSS will support the creation of BT profiles for both Veterans and non-Veteran BT Claimants to ensure efficient processing and payment.

- **Support for Multiple User Interfaces and Modes**

The BTSSS will be accessible via the following user interfaces:

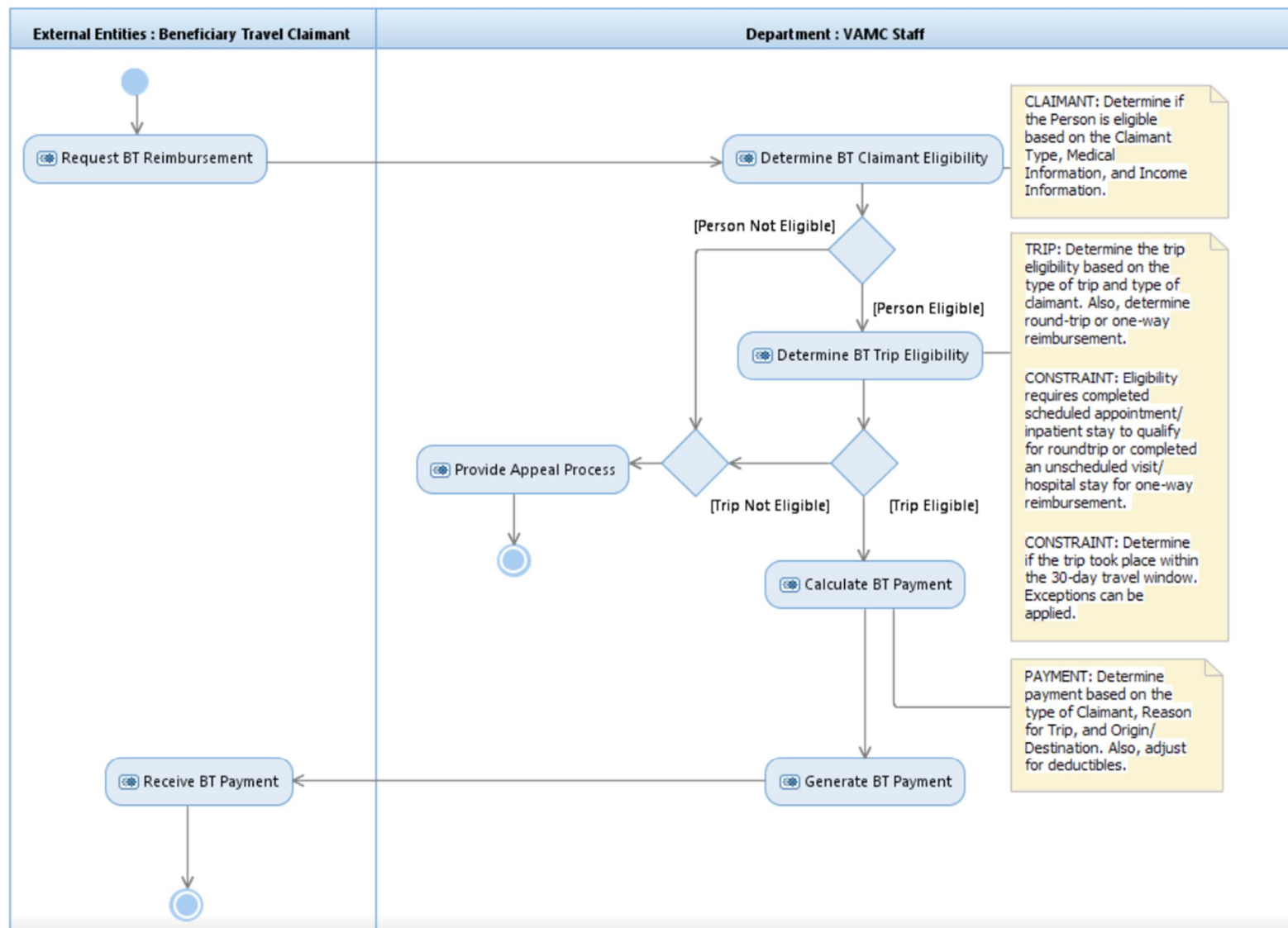
- A standard desktop web browser (for example, PC Desktop, laptop, etc.). This is to ensure quick deployment of the solution and immediate use;

- A user interface optimized for mobile devices (for example, mobile browser); and
- A user interface optimized for use on VA VPS Kiosk devices.
- **Support for Submission of Receipts and Non-Receipt Expenses**  
The BTSSS will support the ability to submit expenses that require a receipt (for example, air travel) and expenses that do not require a receipt (for example, mileage expense).
- **Support for Submission of Expense Report**  
The BTSSS will support the ability to submit an expense report composed of expense entries with scanned images of the supporting documentation. The supporting documentation is used to document expenses for meals/lodging, other expenses (for example, tolls), other transportation modes (for example, air travel), and special mode, for example ferry, ambulance or other non-typical means of transportation.
- **Ability to Plan for Future Beneficiary Travel**  
The BTSSS will need to provide the BT claimant the ability to plan for future trips. This would entail allowing the BT claimant to submit information for a particular future appointment and receive information from the BTSSS about how the claim may be adjudicated (for example, the eligibility of the trip for BT reimbursement and, if eligible, the estimated reimbursement amount).
- **Business Intelligence and Reporting**  
The BTSSS will need to include the ability to produce a range of reports to ensure that the solution is effective and meets the goals and objectives of the Business Owners.

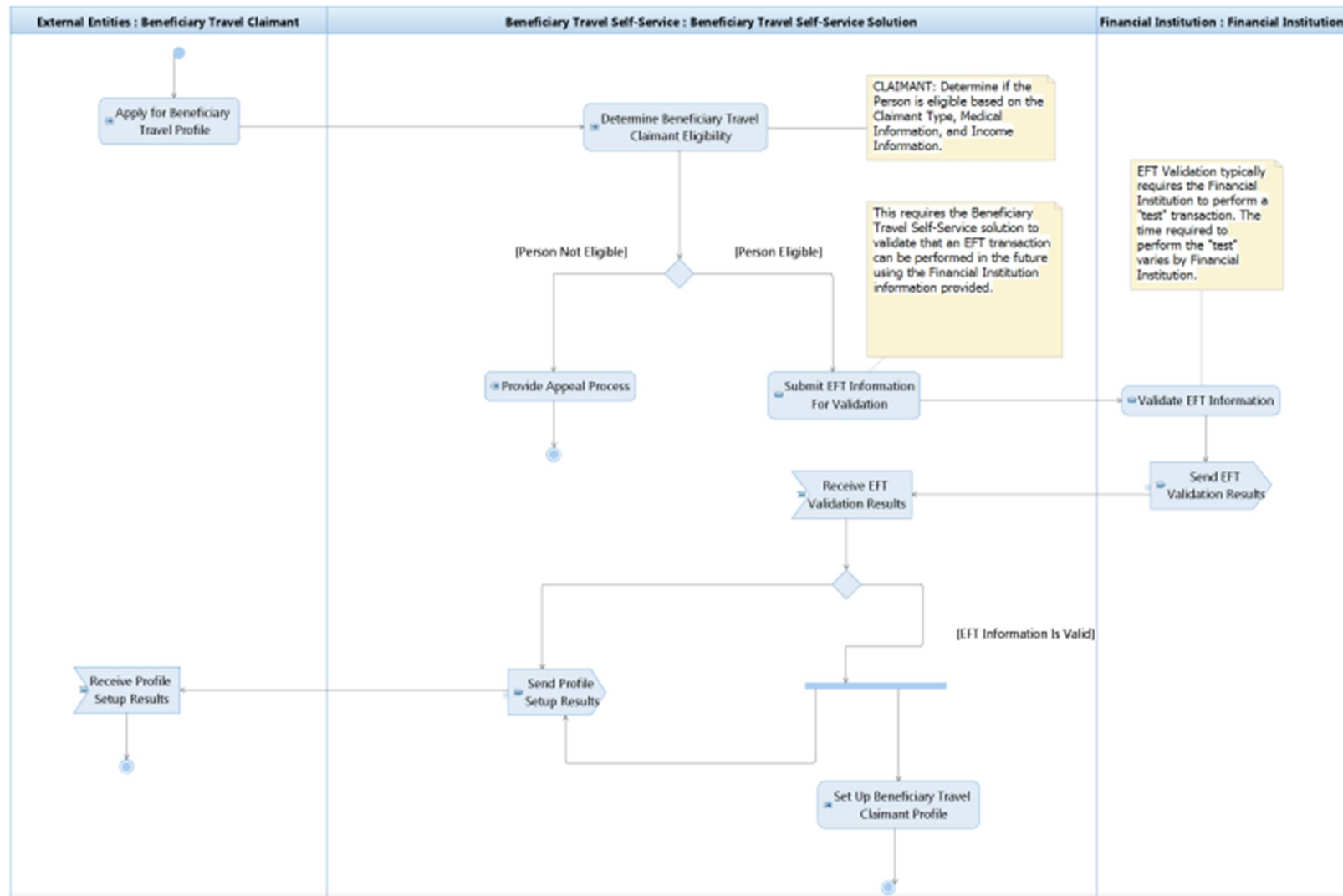
## 2.2. Overview of the Significant Requirements

For a complete list of the requirements, please refer to the [BTSSS Requirements Traceability Matrix \(RTM\)](#).

Figure 1 illustrates the “AS-IS” business process and Figure 2 illustrates the “TO-BE” business process for the BTSSS.



**Figure 1: AS-IS Beneficiary Travel Self-Service System Solution**



**Figure 2: TO-BE Process Beneficiary Travel Profile Application – High-Level View**

## 2.2.1. Overview of Significant Functional Requirements

The table below highlights the significant functional requirements for the BTSSS.

**Table 3: Significant Functional Requirements**

ID	Requirement
BN1	The BTSSS will adhere to the Enterprise-level requirements within the Requirements Management Repository (RMR) and as specifically addressed in Appendix D of the Business Requirement Document (BRD).
BN2	Apply for Beneficiary Travel Profile – The Beneficiary Travel Self-Service System solution will need to provide the Beneficiary Travel Claimant the ability to Apply for Beneficiary Travel Profile.
BN3	Set Up Beneficiary Travel Claimant Profile - If the Beneficiary Travel Claimant application for Beneficiary Travel is approved the Beneficiary Travel Self-Service System solution will set up the Beneficiary Travel Claimant profile.
BN4	Request Beneficiary Travel Reimbursement – The Beneficiary Travel Self-Service System solution will provide the Beneficiary Travel Claimant the ability to request Beneficiary Travel reimbursement.
BN5	Submit Expense Report - The Beneficiary Travel Self-Service System solution will allow the Beneficiary Travel Claimant to create and submit an expense report. An expense report will be required for claims that require receipts and/or other supporting documentation attached in the form of scanned images.
BN6	Process Manual Claims - The Beneficiary Travel Self-Service System solution will provide the capability for the Travel Claimant to manually process a claim. This is required for claims that cannot be processed automatically by the Beneficiary Travel Self-Service System solution.
BN7	Determine Beneficiary Travel Claimant Eligibility – The Beneficiary Travel Self-Service System solution will determine the eligibility of the Beneficiary Travel Claimant using prescribed regulations and guidelines.
BN8	Determine Beneficiary Travel Trip Eligibility – The Beneficiary Travel Self-Service System solution will determine the eligibility of the Beneficiary Travel Claimant's trip using prescribed regulations and guidelines.
BN9	Calculate Beneficiary Travel Payment – The Beneficiary Travel Self-Service System solution will calculate the payment for the Beneficiary Travel Claimant using prescribed regulations and guidelines.
BN10	Apply Payment Adjustments – The Beneficiary Travel Self-Service System solution will apply any required adjustments to the payment for the Beneficiary Travel Claimant using prescribed regulations and guidelines.



ID	Requirement
BN11	Generate Beneficiary Travel Payment – The Beneficiary Travel Self-Service System solution will ensure that a payment is generated only when payment criteria is met, and will forward the request for payment to the correct party (for example, Fiscal Department).
BN12	Receive Beneficiary Travel Payment – This would cover activities related to ensuring that the Beneficiary Travel Claimant receives the payment.
BN13	Provide Appeal Process – The Beneficiary Travel Self-Service System solution will provide the Beneficiary Travel Claimant with the appropriate appeals process depending on the issue encountered and as prescribed by regulations and guidelines.
BN14	Plan Future Beneficiary Travel – The Beneficiary Travel Self-Service System solution will need to provide the Beneficiary Travel Claimant the ability to plan for future trips and submit these as claims.
BN15	Business Intelligence and Reporting – Business Intelligence and Reporting affects the ability to implement and manage all of the capabilities. Business Intelligence and Reporting tools are also a critical component to ensure that the system meets efficiency and costs requirements.
BN16	Automated Workflow Management – The Beneficiary Travel Self-Service System solution will need to include Automated Workflow features (for example, status tracking, notifications). Automated Workflow features are a critical component to ensure that claims are processed within the specified objectives.
BN17	Business Rules Management – The Beneficiary Travel Self-Service System solution will need to include Business Rules management features. Business Rules management features are a critical component to ensure that the solutions allow for flexible changes to rules and guidelines.
BN18	Access Beneficiary Travel Self-Service System Via Standard Web Browser – The Beneficiary Travel Self-Service System will need to be accessible via standard browsers, as specified by Office of Information and Technology (OI&T) standards. In addition, the Beneficiary Travel Self-Service System solution will need to be integrated into existing VA Web Sites.
BN19	Access Beneficiary Travel Self-Service System Via Mobile Device – The Beneficiary Travel Self-Service System will need to be accessible via mobile devices as specified by OI&T standards. In addition, the Beneficiary Travel Self-Service System solution will need to be integrated into existing VA Web Sites.
BN20	Access Beneficiary Travel Self-Service System Via Kiosk – The Beneficiary Travel Self-Service System will need to be accessible via Kiosks as specified by OI&T standards. In addition, the Beneficiary Travel Self-Service System solution will need to be integrated into existing VA Web Sites.

## 2.2.2. Overview of Functional Workload/Performance Requirements

The table below highlights the BTSSS Functional Workload/Performance Requirements.

**Table 4: Functional Workload/Performance Requirements**

ID	Requirement
PFM01	The BTSSS response times provide a visible response to the user (for example,, cursor change to hourglass or data being modified on the screen) that activity is occurring within one second of user input (for example,, mouse click) 95% of the time and within two seconds of user input 99% of the time.
PFM03 PFM04	The BTSSS shall support 1.5 million initial users, capable of expanding to support at least 5% more users, with a target of 10% increase in users per year, and be available on a 24 x 7 basis.
PFM07 PFM08 PFM09	The BTSSS shall support 4.5 million claims per year initially. The BTSSS shall be scalable to support 10 million claims per year. At peak periods, the BTSSS shall be capable of processing 25,000 claims an hour.
PFM10	The BTSSS shall be capable of performing a minimum of 157 instances of CRUD operations on VistA source systems.

## 2.2.3. Overview of Operational Requirements

The table below highlights the operational requirements for the BTSSS.

**Table 5: Operational Requirements**

ID	Requirement
REL01 REL02	The BTSSS maintenance shall be scheduled during off-peak hours (4PM–9AM EST) or in conjunction with relevant VistA maintenance schedules. The BTSSS solution will need to support minimal troubleshooting and maintenance, as defined by the VA for this type of field-deployed system.
REL12	The BTSSS shall require no more than 8 hours of maintenance per month. Servicing and maintenance is expected to occur on the weekends during periods of least usage.
REL03 REL04 REL08	Information about response-time degradation resulting from unscheduled system outages and other events that degrade system functionality and/or performance shall be disseminated to the user community by the standard notification protocol defined by VA. When/if lapses in system/update availability occur, the BTSSS shall display a detailed message including approximate time to come back. The notification shall include the information described in the current Automated Notification Reporting (ANR) template maintained by the VA Service Desk. The business impact must be noted.
REL07	Notification of scheduled maintenance periods that require the service to be offline or that may degrade system performance shall be disseminated to the internal user community (for example, Travel Clerks, Business Users, Application Users, and System Administrators) prior to the scheduled event by the standard notification protocol defined by VA.



ID	Requirement
REL05	Provide a real-time monitoring solution during the maintenance windows.
REL06 SEC05	The BTSSS shall provide periodic preventative backups every 7 days. Data protection measures, such as backup intervals and/or redundancy, shall be consistent with systems categorized as Semi-critical.

## 2.2.4. Overview of Pivotal Technical Requirements

The table below highlights the pivotal technical requirements for the BTSSS.

**Table 6: Pivotal Technical Requirements**

ID	Requirement
Refer to BTSSS RSD Section 2.10	The BTSSS shall follow VA coding practices guidelines. The system shall be compatible with the standards defined in the TRM (Technical Reference Model).

## 2.2.5. Overview of the Security and Privacy Requirements

The table below highlights the security and privacy requirements for the BTSSS.

**Table 7: Security and Privacy Requirements**

ID	Requirement
SEC01 SEC10	The BTSSS shall utilize VA's Active Directory for Travel Clerk authentication. The BTSSS shall utilize VA's Active Directory for Beneficiary Travel internal users (for example, Travel Clerks, Business Users, Application Users, and System Administrators) access control.
SEC06	Ensure the BTSSS security requirements meet all VHA Security, Privacy, and Identity Management requirements, including VA Handbook 6500.
ENTR25	The Security Categorization will drive the initial set of minimal security controls required for the information system. Minimum security control requirements are addressed in National Institute of Standards and Technology (NIST) SP 800-53 and VA Handbook 6500, Appendix D.
ENTR99	All VA security requirements will be adhered to. Based on Federal Information Processing Standard (FIPS) 199 and NIST SP 800-60, recommended Security Categorization is High.
ENTR10	All VA Privacy requirements will be adhered to. Efforts that involve the collection and maintenance of individually identifiable information must be covered by a Privacy Act system of records notice.

## 2.2.6. Overview of System Criticality and High Availability Requirements

The table below highlights the system criticality and high availability requirements for the BTSSS.

**Table 8: System Criticality and High Availability Requirements**

ID	Requirement
REL09	The BTSSS solution shall be available from 6AM to 11PM Eastern Standard Time (EST), 7 days a week, and 365 days a year.
REL10 REL11	During this time window, the BTSSS shall maintain operational availability (Ao) of 99% from 6AM to 11PM EST. Outside of this time window, the BTSSS solution shall maintain Ao 95% of the time.

## 2.2.7. Overview of Single Sign on Requirements

The table below highlights the single sign on requirements for the BTSSS.

**Table 9: Single Sign-On Requirements**

ID	Requirement
SEC02	The BTSSS Travel Clerk authentication shall be provided through Single Sign-On (SSO) with the existing VA Microsoft Active Directory servers.
SEC06	Ensure the proposed solution meets all VHA Security, Privacy and Identity Management requirements, including VA Handbook 6500 and VA Handbook 6500.6 Contract Security.

## 2.2.8. Overview of Enterprise Portals Requirements

The table below highlights the use of Enterprise Portals for the BTSSS.

**Table 10: Enterprise Portals Requirements**

ID	Requirement
OWNR 3.03	The BTSSS will need to interact with other VA systems and interfaces, including but not limited to: the Financial Management System (FMS), Enrollment System Redesign (ESR), Master Veteran Index (MVI) and Patient Care Encounter (PCE).
OWNR 19.02 OWNR 19.03	The BTSSS will need to be available from within MyHealtheVet and eBenefits.
REL13	The BTSSS shall ensure 100% synchronization for data extracted from systems of record.

## 2.2.9. Overview of Special Device Requirements

No special device requirements have been identified at this time.

### 3. Conceptual Design

This section will be updated when the software developer has been identified and has begun development agile sprint cycles.

*This section of the SDD provides details about the following topics:*

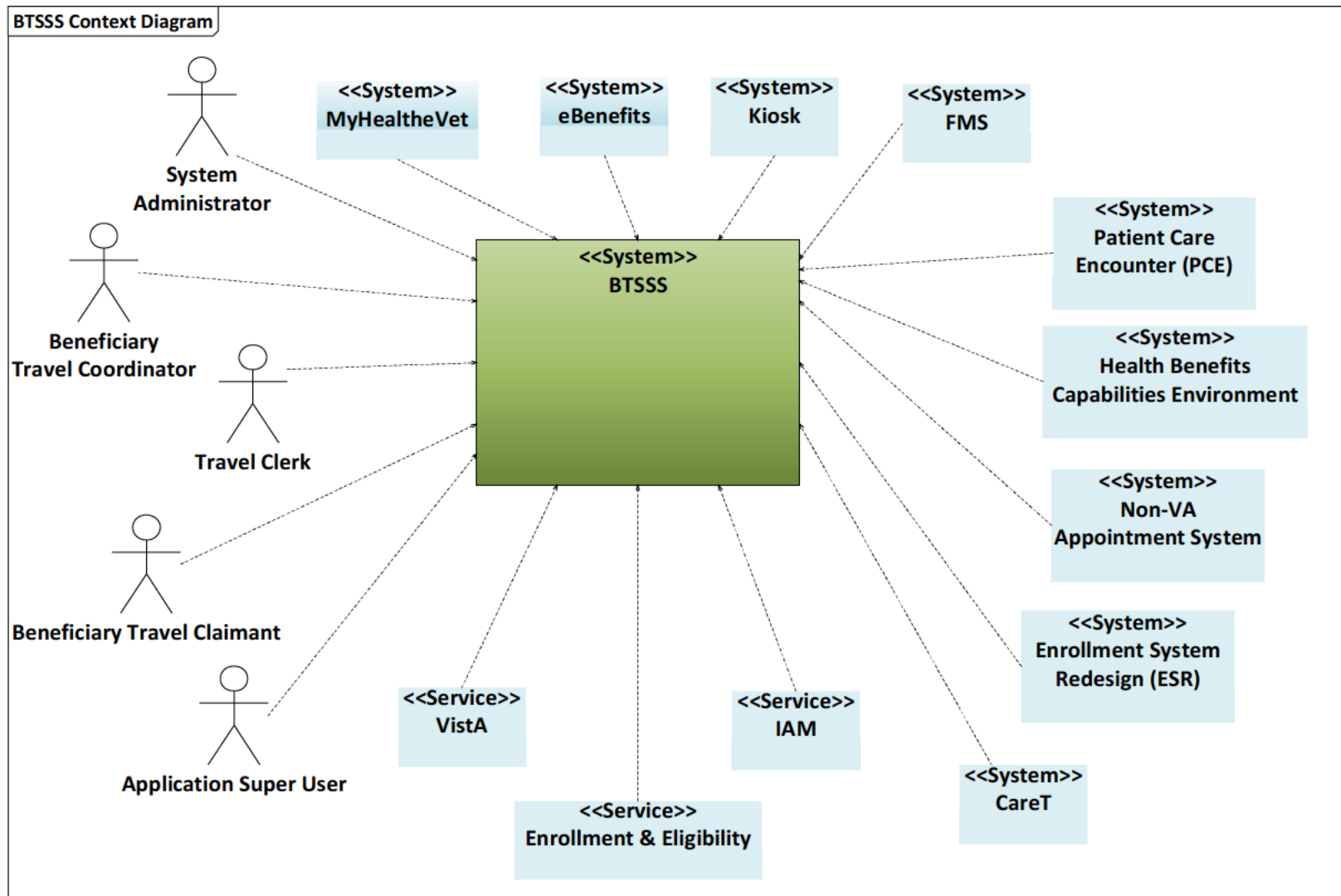
- *Conceptual Application Design*
- *Conceptual Data Design*
- *Conceptual Infrastructure Design*

#### 3.1. Conceptual Application Design

*Provide details of the ‘As-is’ view of the existing system along with the design of “the current increment” and the “To-be.”. The “To-Be” view should include the future application context, and application high level design. The “current increment” view should have application context and high level design of this specific increment that this SDD addresses.*

##### 3.1.1. Application Context

The application context describes the system as a whole and allows us to focus on the external factors and events that impact the BTSSS. The BTSSS will have inputs and output from existing services and system.



**Figure 3: Sample Application Context Diagram**

Table 11 describes the information in the Application Context Diagram in four sections. Note that the system for which this design applies is represented by a single object (typically in the center of the diagram). Therefore, it is not referred to in Table 11 below.

**Table 11: (Object Grouping): Application Context Description**

<b>ID</b>	<b>Name</b>	<b>Description</b>	<b>Interface Name</b>	<b>Interface System</b>
<<System>> MyHealtheVet	MyHealtheVet	One of the many VA portals for Veteran's personal health record.	BTSSS/MyHealtheVet	MyHealtheVet
<<System>> eBenefits	eBenefits	VA portal for Veteran benefits information and management	BTSSS/eBenefits	eBenefits
<<System>> Kiosk	VPS Kiosk	Standalone systems at various VAMCs that allow users to access different VA	BTSSS/Kiosk	Kiosk
<<System>> Financial Management System (FMS)	Fiscal Department	This system, or group of systems, is responsible for the processing and distribution of travel reimbursement payments to	BTSSS/Payment Source (FMS or Other Systems)	FMS
<<System>> Patient Care Encounter (PCE)	Patient Care Encounter	This system is used to manage outpatient encounters and appointment information in VAMC	BTSSS/PCE	PCE
<<System>> Health Benefits Capabilities Environment	Health Benefits Capabilities Environment	One of the many VA portals for Veteran health benefits.	BTSSS/HBCE	HBCE
<<System>> Non-VA Appointment System	Non-VA entities	This object represents appointment systems used by non-VA medical facilities	BTSSS/Non-VA Appointment System	Non-VA Appointment System
<<System>> Enrollment System Redesign (ESR)	Enrollment & Eligibility (E & E)	This object represents a to-be system used by E & E	BTSSS/ESR	E & E
<<System>> CareT	Caregiver Support Program (CSP)	This object represents the to-be system used by CSP to support disabled Veterans and their caregivers	BTSSS/CareT	CareT

ID	Name	Description	Interface Name	Interface System
<<Service>> Identity and Access Management (IAM)	IAM	This object represents the enterprise service that will provide provisioning, credentialing, and identity authentication to the BTSSS	BTSSS/IAM	IAM
<<Service>> Eligibility and Enrollment	E & E	This service provides enrollment/eligibility information.	BTSSS/ Eligibility and Enrollment	E & E
<<Service>> VistA	VistA	This service is an enterprise-wide information service used throughout the VA medical system	BTSSS/ VistA	VistA

**Table 12: Interfaces External to OI&T**

ID	Name	Related Object	Input Messages	Output Messages	External Party
1	BTSSS/Non-VA Appointment System	<<System>> Non-VA Appointment System	Appointment/Encounter Information	Information Requests	Non-VA entities
2	BTSSS/FMS	<<System>> Financial Management System (FMS)	Payment Response File	EFT Information  Payment Information File	Fiscal Department

**Table 13: Interfaces Internal to OI&T**

ID	Name	Related Object	Input Messages	Output Messages	External Party
1	BTSSS/MyHealtheVet	<<System>> MyHealtheVet	Claimant Information	N/A	MyHealtheVet
2	BTSSS/eBenefits	<<System>> eBenefits	Claimant Information	N/A	eBenefits
3	BTSSS/Kiosk	<<System>> Kiosk	Claimant Information	N/A	Kiosk
4	BTSSS/PCE	<<System>> Patient Care Encounter (PCE)	Medical Appointment Status and Encounter Information	Information Requests	PCE

ID	Name	Related Object	Input Messages	Output Messages	External Party
5	BTSSS/HBCE	<<System>> Health Benefits Capabilities Environment	Claimant Information	Request for Claimant's Information	Health Benefits Capabilities Environment
6	BTSSS/ESR	<<System>> Enrollment System Redesign (ESR)	Claimant Enrollment Information	Request for E & E Information	Enrollment & Eligibility (E & E)
7	BTSSS/CareT	<<System>> CareT	Claimant's Caregiver Status	Request for Caregiver's Status	Caregiver Support Program (CSP)
8	BTSSS/IAM	<<Service>> Identity and Access Management (IAM)	User Identity	User Access Request	IAM
9	BTSSS/ Eligibility and Enrollment	<<Service>> Eligibility and Enrollment	Claimant Enrollment Information	Request for E & E Information	E & E
10	BTSSS/ VistA	<<Service>> VistA	Medical Record  Appointment Information	N/A	VistA

**Table 14: Externally Shared Data Stores**

ID	Name	Data Stored	Owner	Access
<i>&lt; ID from diagram &gt;</i>	<i>&lt;Name of the data store&gt;</i>	<i>&lt;Description of the data being stored&gt;</i>	<i>&lt;This System / Name of OIT or external organization&gt;</i>	<i>&lt;Enter the Create, Read, Update, or Delete (CRUD) operations that this system does on this data store&gt;</i>



### 3.1.2. High-Level Application Design

The BTSSS will most likely consist of three layers – Presentation, Business Logic, and Data. These layers will encapsulate the various modules and components of BTSSS. The designer will formulate BTSSS modules and components in these layers.

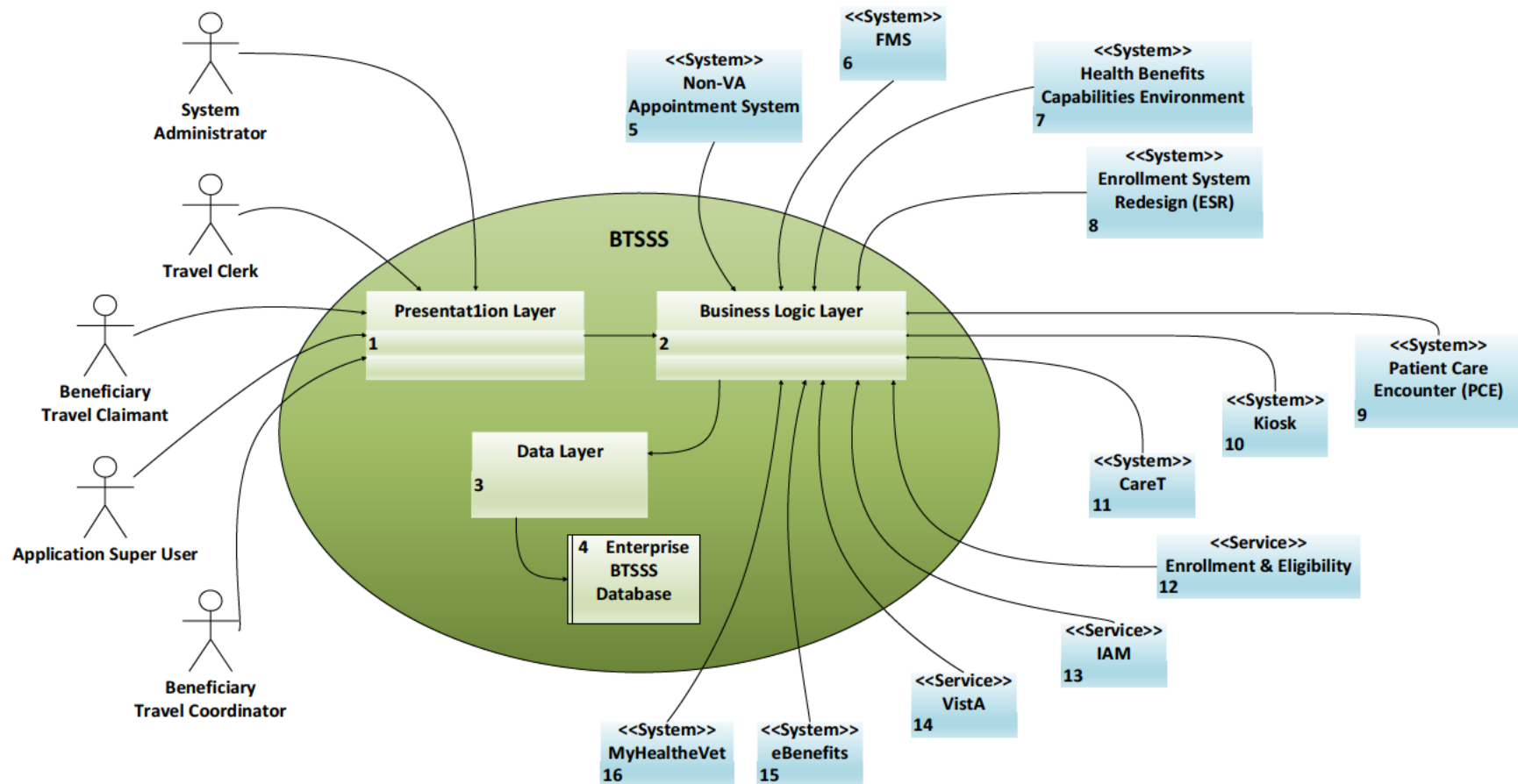


Figure 4: BTSSS High-Level Application Design



**Table 15: Objects in the High Level Application Design - Objects/Components to be Built or Modified**

ID	Name	Description	Service or Legacy Code	External Interface Name	External Interface ID	Internal Interface Name	Internal Interface ID	SDP Sections 1&2
1	Presentation Layer	This layer provides BTSSS with the ability to interact with its intended users.	n/a	System Administrator Travel Clerk Beneficiary Travel Claimant Application Super User Beneficiary Travel Coordinator	System Administrator Travel Clerk Beneficiary Travel Claimant Application Super User Beneficiary Travel Coordinator	Presentation /Business Logic	Presentation /Business Logic	
2	Business Layer	This layer encapsulates BTSSS business rules and work flows	n/a	MyHealtheVet eBenefits VistA IAM E & E CareT Kiosk PCE ESR HBCE FMS Non-VA Appt. System	MyHealtheVet eBenefits VistA IAM E & E CareT Kiosk PCE ESR HBCE FMS Non-VA Appt. System	Presentation /Business Logic Business Logic/Data	Presentation /Business Logic Business Logic/Data	
3	Data Layer	This object provides BTSSS with the ability to process, store, and share data	n/a	Enterprise BTSSS Database	Enterprise BTSSS Database	Business Logic/Data	Business Logic/Data	

**Table 16: Internal Data Stores**

ID	Name	Data Stored	Steward	Access
4	Enterprise BTSSS Database	This database contains Beneficiary's Personal Identifiable Information (PII), financial information, and travel plan information.	BTSSS	CRUD

### 3.1.3. Application Locations

*Use Table 17 to specify the locations at which the application components will be hosted.*

*Consideration should be given to adopt cloud technologies as potential solutions. Leveraging cloud technologies is part of a larger effort by the Office of Management and Budget (OMB) to reform Federal IT Management. Considerations such as regional deployments etc. should be documented in this section.*

**Table 17: Application Locations**

Application Component	Description	Location at Which Component is Run	Type
<Component name>	<Description>	<Facility name>	<Presentation Logic/Business Logic/Data Logic/Interface Code>

**Table 18: Application Users**

Application Component	Location	User
<Component name>	<Facility name>	<Role>

## 3.2. Conceptual Data Design

This section will be updated when the software developer has been identified and has begun development agile sprint cycles.

### 3.2.1. Project Conceptual Data Model

*A project conceptual data model (CDM) is a high-level representation of the data entities and their relationships. It does not normally include the data elements that comprise each entity. It is a first step toward developing the more detailed logical data model (LDM) that will be provided during the Logical Data Design.*

*Figure 5 illustrates a sample of a project CDM.*

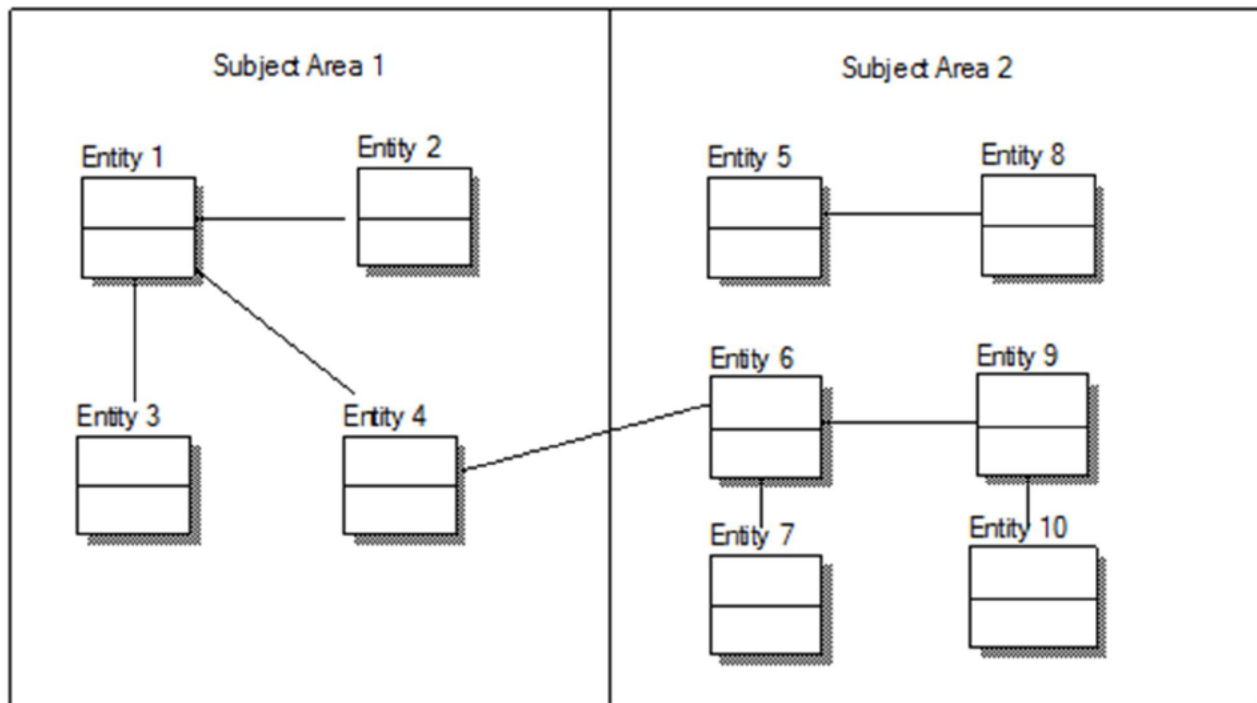


Figure 5: Sample Project Conceptual Data Mode

### 3.2.2. Database Information

Use Table 19 to identify all the databases that will be created, replaced, interfaced with, or whose structure will be modified (i.e., add or delete tables or add or delete columns to a table) as part of this effort.

Table 19: Database Inventory

Database Name	Description	Type	Steward
<Name>	<Description>	<Create/Replace/Interface/Modify>	<Application/Organization that is the steward>

### 3.2.3. User Interface Data Mapping

This section describes and defines the format and information that will be available for users of the product to be able to enter data into the database or to retrieve information from the database, if applicable.

#### 3.2.3.1. Application Screen Interface

Create a new subsection for each screen of the Graphical User Interface (GUI) that users will have access to, in order to enter or update information in the database.)

##### 3.2.3.1.1. <Insert name of screen>

Figure 6: <screen name> Screen represents the screen that <describes what the screen accomplishes>; Table 20 describes it. Paste a screenshot below and complete the table to describe the screen.

To be added when available

**Figure 6: <screen name> Screen**

**Table 20: <screen name> Screen Description**

Graphical User Interface (GUI) Field	Table (Database Table that field connects to)	Field (Field in Table that the GUI field connects to)	Comments
<Name>	<xxx>	<PATIENT_NAME>	<Add any comments or descriptive information that would be relevant to the tester>
<SSN>	<xxx>	<SSN>	
Date of Birth (Age)	<yyyy>	DATE_OF_BIRTH DATE_OF_DEATH (if deceased)	

### 3.2.3.2. Application Report Interface

*This section describes and defines the reports that will be available in the user interface, if applicable.*

#### 3.2.3.2.1. <Insert name of report>

*<Create a new subsection for each report> Figure 6 represent <name> screen and Table 20 describes it...*

*Figure 7 represents the <report name>; Table 21 describes it. Paste a screenshot of the report below and complete the table to describe the report.*

To be added when available

**Figure 7: <Report name> Report**

**Table 11: <Report name> Description**

Report Column	Data Source <Table Name. Fieldname>
Patient	<xxx.PATIENT_NAME>
SSN	<xxx.SSN>
DoB	<yyy.DATE_OF_BIRTH>

### **3.2.3.3. Unmapped Data Element**

*In this section describe any database element that was not mapped to a screen and the reason the data element(s) was not mapped. This section may be skipped if there is no User Interface involved in the project, such a building a service offering etc.*

## **3.3. Conceptual Infrastructure Design**

*The Conceptual Infrastructure Design should describe any unique technology that will be used, which are either part of this system, or will attach to this system.*

*. Because the system is at a preliminary design stage, it is expected that the information provided may need to be changed during later design stages or increments.*

*The Conceptual Infrastructure Design is a high-level overview of the infrastructure that will be used to support the application. Primary emphasis is on the environments that will be required and the locations at which they will be installed. The Conceptual Infrastructure Design becomes more detailed at later stages as more information is collected regarding the system, and the infrastructure requirements (i.e., capacity requirements) are better known.*

### **3.3.1. System Criticality and High Availability**

The BTSSS RSD calls for an operational availability (Ao) of 99% from 6AM to 11PM Eastern Standard Time, and an Ao of 95% during all other hours. The BTSSS is to maintain this operational availability year-round, with the exception of maintenance. The Mean Time To Repair (MTTR), as specified by the Requirement Specification Document, is less than 24 hours.

To achieve this goal, BTSSS may utilize several design strategies:

- 1) Redundancy in both hardware components and software processing modules
- 2) Conduct fault-tree analysis and determine potential failure modes for BTSSS
- 3) Reuse of implemented service/code/database

Redundancy enables BTSSS to perform its functions in the event of error/failure in one or more parts of the system. This applies to both hardware and software of BTSSS. Hardware that supports BTSSS (workstations, servers, etc.) should be redundant to guard against failure of critical equipment. The same logic applies to the software portion of BTSSS as well. Software components that perform critical BTSSS system functions should also have redundant or alternative pathways to complete these functions.

In order to mitigate the possibility of software failure, fault-tree analysis should also be conducted to identify potential failure modes of BTSSS. These potential failure modes should be tracked and mitigated as the system development progresses.

Finally, reuse of implemented services/code/databases should be done to minimize development time and integration difficulty. BTSSS should reuse services, database, and other software/hardware components as much as feasible.

In addition to system availability, BTSSS will follow VA guidance on contingency planning and disaster recovery. Detailed specifications can be found in the BTSSS RSD, Section 2.4, Disaster Recovery Specification.

### 3.3.2. Special Technology

*If any special technology was identified in Section 2.5.9 as part of this system, describe the device and the type of location at which it will be installed. This information may be provided using Table 21.*

**Table 21: Special Technology Requirements**

Special Technology	Description	Notional Location	TRM Status
<Name>	<Business language description>	<At what type of location will this technology be deployed?>	<Is this technology in the TRM? (Yes / No)>

### 3.3.3. Technology Locations

*This section describes the various technology components that will be used. If known, provide the name of the datacenter at which the technology will be installed. If not, specify as Site A, Site B etc. Provide this information in Table 22.*

**Table 22: (Grouping) Technology Location Details**

Technology Component Production 1	Location	Usage
Workstations		
Special Hardware		
Interface Processors		
Legacy Mainframe		
Legacy Application Server		
Legacy Databases		
Other		

Technology Component Production 2	Location	Usage
<copy from Prod 1 set, or enter new ones as appropriate>		

Technology Component Certification	Location	Usage

Technology Component Education	Location	Usage

Technology Component Test	Location	Usage

Technology Component Development	Location	Usage

### 3.3.4. Conceptual Infrastructure Diagram

#### 3.3.4.1. Location of Environments and External Interfaces

*Create a diagram to show the environments that will be supported. As illustrated in the diagram should show the following:*

- *Local networks to which they will be attached (Production, Test, or Development)*
- *Locations at which they will be installed*
- *External connections (each external interface should be shown in terms of where it enters the network).*

**Figure 8: Sample Conceptual Networks and Environments**

#### 3.3.4.2. Conceptual Production String Diagram

*Create a diagram to show the configuration of a single production string.*

*Additional components, such as the mainframe, other Web servers, or other major components should be included if they are expected to be required.*

**Figure 9: Conceptual Production String Diagram**



## 4. System Architecture

This section describes BTSSS hardware, software, and network architectures. In addition, this section will also describe BTSSS in relation to Enterprise Shared Services and VA's Enterprise Technical Architecture.

### 4.1. Hardware Architecture

BTSSS will be virtually hosted in the AITC. The high-level hardware architecture listed below represents the typical hardware set-up for virtually hosted systems at AITC. The developer may revise the hardware architecture at a later date, pending VA approval.

BTSSS capacity and resource needs fluctuate throughout the day; virtually hosting at AITC (or other VA technical centers) enables dynamic resource allocation (storage, processing power, memory, etc.). Resource allocation can be arranged based on demand. As demand for BTSSS grows during peak hours, processing power and memory from server(s) can be temporarily allocated to BTSSS.

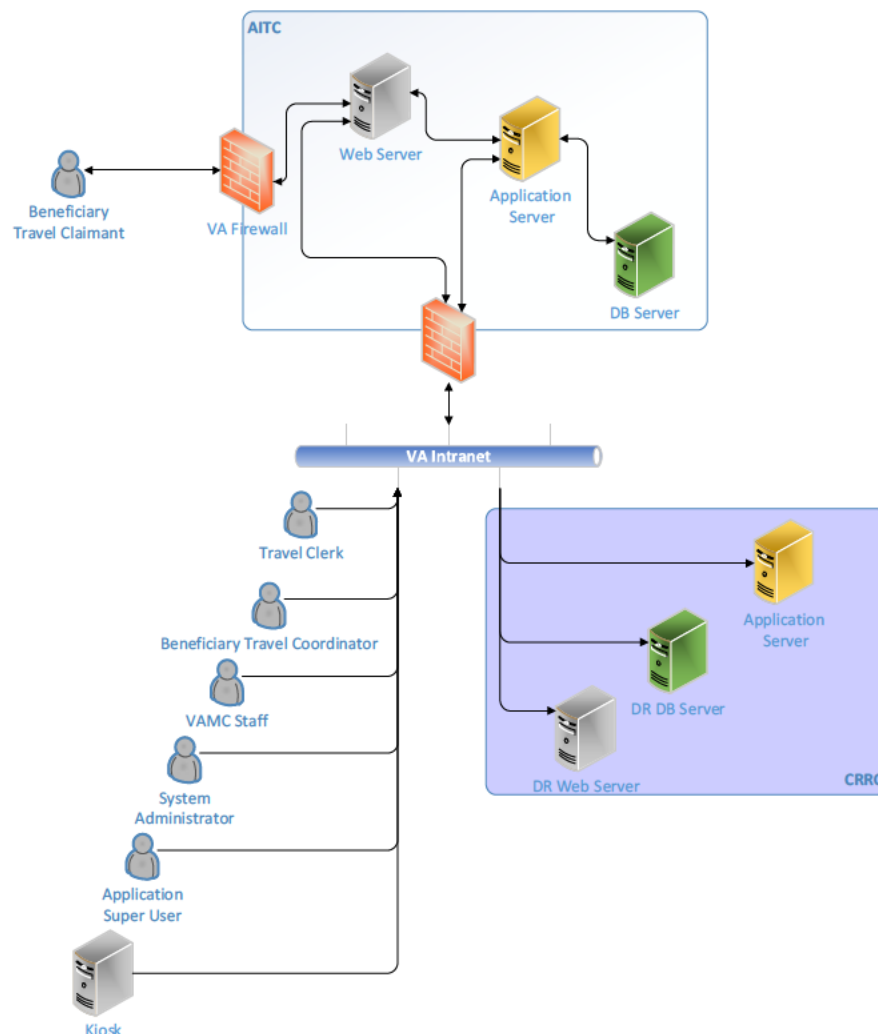


Figure 10: BTSSS Hardware Architecture



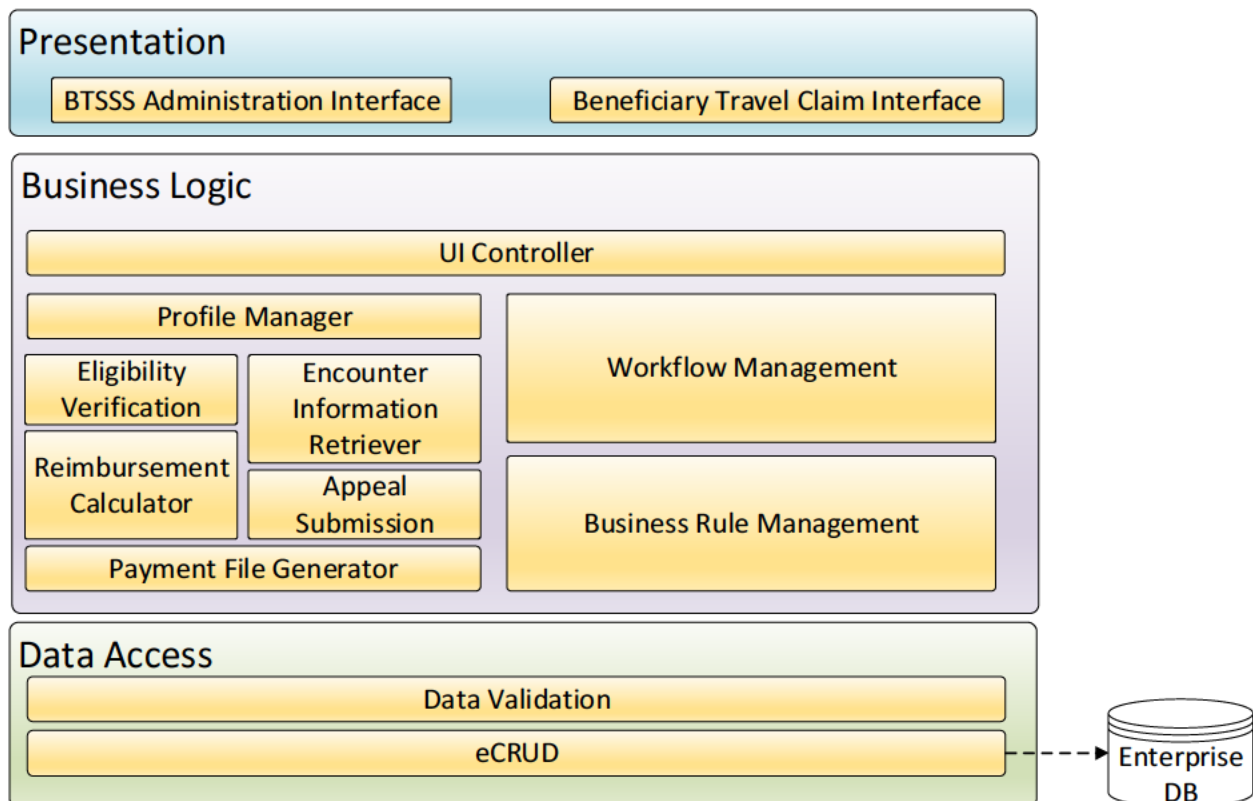
BTSSS hardware will comprise application server(s), database server(s), and web server(s). Application server(s) is where business logics are executed. Web server(s) is the portal in which BTSSS will interact with web. Database server(s) is responsible management of information use in various VA business processes.

VA users and machines will access BTSSS via the VA intranet. External users, such as the Beneficiary Travel Claimant, will access BTSSS from his/her existing internet connection.

## 4.2. Software Architecture

The purpose of this section is to clearly lay out what a functionally correct BTSSS should look like. The generic components are modeled based on the functional requirements outlined in the RSD and the RTM. Tracability between software components and functional requirements is outlined in Appendix A.4.

The BTSSS software architecture is modeled after the N-tier architecture pattern outlined in [VA EA's ETA Compliance Criteria](#). This architectural pattern is based on the well-worn software systems engineering principle of separation of concerns. Layers within this architectural pattern are loosely coupled such that one layer can be modified without the extensively changing others. The BTSSS layered architecture consists of three main layers – the Presentation, Business Logic, and Data Access layers. These layers are further decomposed into various generic components, as depicted below.

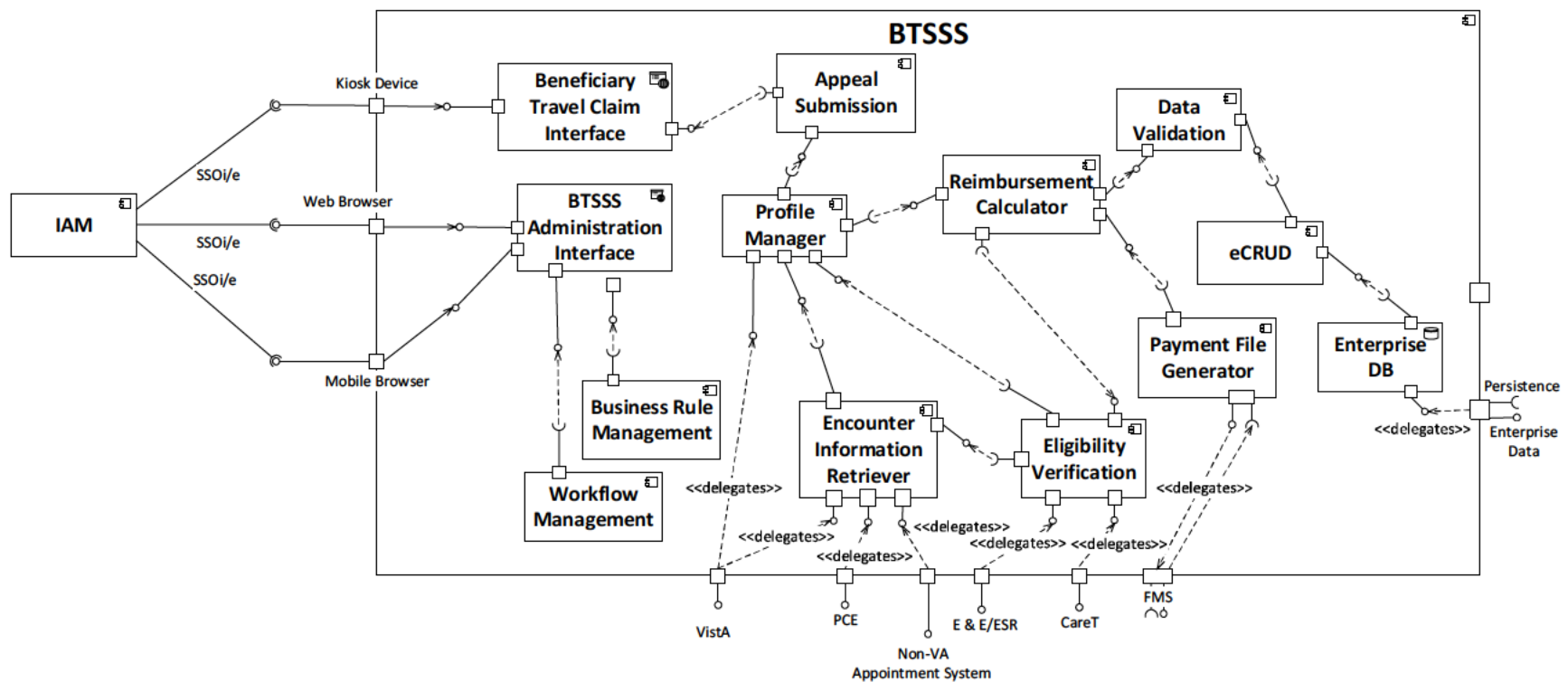


**Figure 11: BTSSS Software Architecture**

In congruence with Section 2.4.12 of the ETA Compliance Criteria document, BTSSS will adopt the “thin-client” approach. This means that the Presentation layer contains only the BTSSS user interfaces, whereas the business logic of BTSSS will reside in another layer. The user interfaces have two archetypes/components. One is the Beneficiary Travel Claim interface. This interface will be the primary user interface for Beneficiary Travel Claimants, Travel Clerks, VAMC staff members, and Beneficiary Travel Coordinators. The BTSSS Administration user interface is the primarily for Application Super Users, System Administrators, Business Users, and Product Owners. These user interfaces shall be designed for web browsers outlined in the VA Technical Reference Model (TRM).

The Business logic layer contains the core components of the BTSSS. The components within this layer implement functionalities that satisfy the business owner’s needs, as documented by the BRD and the RSD. In general, these modules/components will implement profile creation/management, claim processing, reimbursement calculation, payment generation, work flow management, and business rule management functions. Claim Processing functions include eligibility verification for the claimant and the trip, patient encounter information retrieval and verification, and claim appeals. Reimbursement calculation includes functions such as travel mileage calculation and reimbursement calculation. Payment generation includes functions such as automatic FMS payment file generation and reimbursement payment validation.

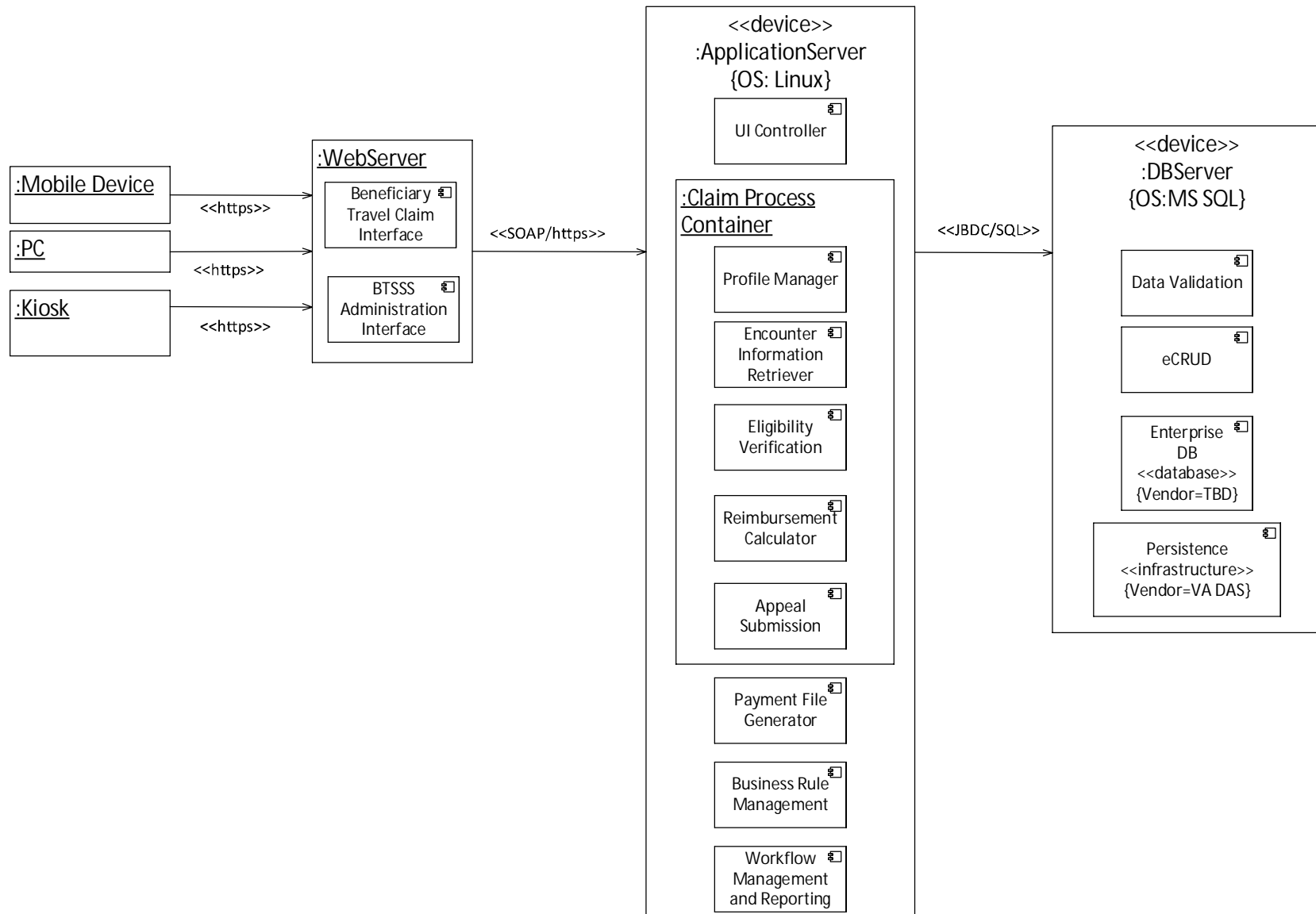
The Data Access layer will be responsible for data persistence, data validation, data management, and CRUD (Create/Read/Update/Delete) operations requested by various components and external systems. In addition, the Data Access layer will be responsible for the retrieval (via various protocols) of information from both VA and external systems.



**Figure 12: BTSSS Component Diagram**

The generic components depicted in this component diagram represent a high level, white-box view of BTSSS. Step-wise refinement is needed to further establish the composition of these components.

The Deployment Diagram below depicts how these generic components can be feasibly implemented in the hardware architecture outlined in Section 4.1 of this document.



**Figure 13: BTSSS Deployment Diagram**

### 4.3. Network Architecture

The high-level network architecture consists of multiple tiers of components, as Figure 14 depicts in a generic form. These tiers group into the following categories:

- A front-end user interface/portal tier;
- A middle tier that includes the proxy servers, policy servers, and other components; and
- A back-end tier that consists of user repositories and related components.

The actual network layout and specifics will be defined further after the actual software and development award is issued.

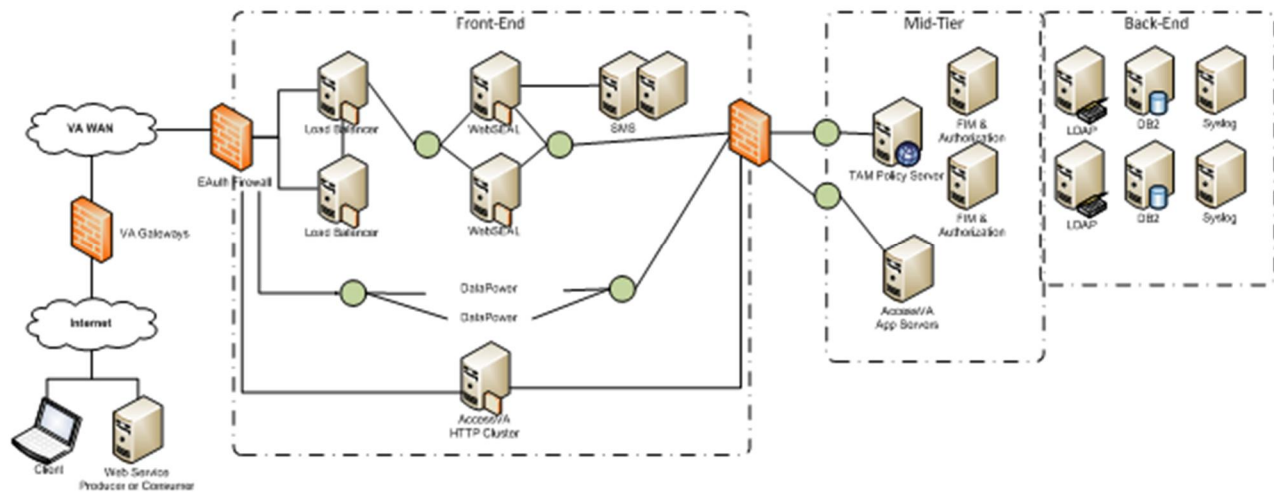


Figure 14: High-Level Network Architecture

### 4.4. Service Oriented Architecture / ESS

The VA does not anticipate that the BTSSS solution will be entirely self-contained, because the defined requirements interfaces with external financial systems and internal interfaces, such as IAM integration, will be required. However, until a solution and solution interface capability are further defined, this section will not have additional information.

### 4.5. Enterprise Architecture

The system design will comply with VA policies, directives, and procedures, including:

- One-VA Technical Reference Manual (TRM);
- VA ProPath and PMAS standards;
- VA Assessment and Authorization (A&A) and related security policies; and
- Open Source VA Code Guidelines.

If the planned system design is found to conflict with a VA policy, directive, or procedure, this conflict will be discussed with the VA Project Manager, who will help determine whether a waiver or exception will be requested with the applicable VA governing board.

## 5. Data Design

*This section outlines the design of the database management system (DBMS) and non-DBMS files associated with the system. For networks, detail the distribution of data and identify any changes to the logical data model that may occur due to software or hardware requirements.*

*Note: Provide a data dictionary appendix showing data element name, type, length, source, validation rules, maintenance, data stores, outputs, aliases, and description.*

### 5.1. DBMS Files

*If a database will be used list and describe the logical requirements that exist for data formats, storage capabilities, data retention, data integrity, etc.*

*Describe how the database will be designed, including the following information, as appropriate:*

- *Logical model; provide normalized table layouts, entity relationship diagrams, and other logical design information*
- *DBMS schemas, subschemas, records, sets, tables, storage page sizes*
- *Access methods (such as indexed, via set, sequential, random access, sorted pointer array)*
- *Estimate the database file size or volume of data within the file, data pages, including overhead resulting from access methods and free space*
- *Definition of the update frequency of the database tables, views, files, areas, records, and sets*
- *Estimates on the number of transactions that the database may have to process.*

### 5.2. Non-DBMS Files

- *Describe all non-DBMS files including narratives on the usage of each file.*
- *Identify if the file is used for input, output, or both; identify temporary files, which modules read and write the file, and similar.*
- *Identify record structures, record keys, indices, and reference data elements within the records.*
- *Define record length and blocking factors.*
- *Define the file access method such as: index sequential, virtual sequential, random access.*
- *Estimate the file size or volume of data within the file.*
- *Define the update frequency of the file if appropriate. Provide the estimated number of transactions per unit time and the statistical mean, mode, and distribution of those transactions.*

## 5.3. Data View

*A "Data View" should be included in the Architectural Representation whenever persistent data objects are included in the system (they are typically present in most software systems). The data view describes the logical data model of the system and includes an Entity Relationship Diagram (ERD). For a description of Entity Relationship diagramming please refer to the whitepaper <[http://www-106.ibm.com/developerworks/rational/library/content/03July/2500/2785/2785\\_uml.pdf](http://www-106.ibm.com/developerworks/rational/library/content/03July/2500/2785/2785_uml.pdf)>*

## 6. Detailed Design

*This section describes the proposed design in detail. Provide the necessary information for the development team to integrate the hardware components and write the software code, so that the hardware and software components will provide a functional product. This is the detailed design, based upon the conceptual design (high level) that was described in the document up to this point.*

*Note: Every design item should map back to the Requirements Specification Document. These should be captured in the Requirement Traceability Matrix (RTM).*

### 6.1. Hardware Detailed Design

*The information requested in this section may be provided by Engineering and/or the Developers. The information provided here is mainly for use by Engineering and Operations.*

*In this section, provide enough information for the developers to build and/or procure the system's hardware. The level of detail requested should be treated as a general guideline and can be omitted if it needs to be filled in by Engineering and Operations.*

*Note: If this section becomes too lengthy, consider incorporating it as an appendix or reference it in a separate document. Add additional diagrams, if necessary, to describe each component and its functions.*

*Include the following information (as applicable):*

- *How much compute capacity? (MFLOPS, TPMs etc.)*
- *System Memory*
- *Local and Shared storage*
- *Network requirements (Bandwidth, Latency etc.)*
- *Public or Private cloud*

### 6.2. Software Detailed Design

*This section provides conceptual and final detailed information associated with the design of the software being delivered. This should be an extension of the corresponding section from Section 3.1, but should contain additional detail as the project progresses.*

#### 6.2.1. Conceptual Design

*This section introduces the conceptual information that establishes the basis for how the software will be built.*

#### **6.2.1.1. Product Perspective**

*This subsection of the SDD should put the product into perspective with other related products. If the product is independent and completely self-contained, it should be stated here. If the SDD defines a product that is a component of a larger system, then this subsection should relate the requirements of that larger system to functionality of the software and should identify interfaces between that system and the software.*

*A block diagram showing the major components of the larger system, interconnections, and external interfaces can be helpful.*

*Sections of the Requirements Specification Document (RSD) can be referenced in the subsections, if applicable.*

##### **6.2.1.1.1. User Interfaces**

*This subsection should specify the logical characteristics of each interface between the software product and its users. This includes those configuration characteristics necessary to accomplish the software requirements (e.g., screens, roll and scroll, GUI interface).*

*Recommendation: Create a block diagram showing the user interfaces.*

##### **6.2.1.1.2. Hardware Interfaces**

*This subsection should specify the logical characteristics of each interface between the software product and the hardware components of the system. This includes configuration characteristics (for example, hardware platform or mainframe versus personal computer). It also covers matters such as what devices the system will support, how they will be supported, and protocols. Examples include scanners, pen driven devices, and radio frequency devices.*

*Recommendation: Create a block diagram showing the hardware interfaces.*

##### **6.2.1.1.3. Software Interfaces**

*This subsection should specify the use of other required software products (e.g., VA Kernel, VA FileMan, Windows NT); and interfaces with other applications or other systems such as commercial off-the-shelf (COTS) or national databases. Specify the application interfaces (e.g., the linkage between an accounts receivable system and a general ledger system and a COTS software package that will be interfaced using an existing interface). This section should provide the following information for each required software product:*

- *Name*
- *Version number*
- *Discussion of the purpose of the interfacing software as related to this software product*
- *Definition of the interface in terms of message content and format (e.g., Health Level Seven [HL7], electronic data interchange).*

##### **6.2.1.1.4. Communications Interfaces**

*This subsection should specify the various interfaces to communications such as local network protocols, e-mail, Transmission Control Protocol (TCP), modems.*

*Recommendation: Create a block diagram showing the communications interfaces.*



#### **6.2.1.1.5. Memory Constraints**

*This subsection should specify any applicable characteristics and limits on memory or partition size.*

#### **6.2.1.1.6. Special Operations**

*This subsection should specify the special operations required by the user such as backup, recovery, and archiving operations.*

*This section should also include any operations for external devices or COTS systems.*

#### **6.2.1.2. Product Features**

*This subsection should provide a summary of the major features of the software.*

*For example, an SDD for an accounting program might use this section to address customer account maintenance, customer statement, and invoice preparation without mentioning the vast amount of detail that each of those features requires.*

*Note: For clarity, remember these items when creating this section of the SDD:*

- *The features should be organized in a way that makes the list of features understandable to the customer or to anyone else reading the document for the first time.*
- *Textual or graphical methods can be used to show the different features and their relationships.*
- *Such a diagram is not intended to show a design of a product, but simply shows the logical relationships among variables.*

#### **6.2.1.3. User Characteristics**

*This subsection should describe the general characteristics of the intended users of the product, including experience and technical expertise. It should not be used to state specific requirements but rather should provide the reasons why certain specific requirements are specified in the RSD.*

#### **6.2.1.4. Dependencies and Constraints**

*This subsection should provide a description of any other items that will limit the developer's options. The following list includes items that limit the developer's options.*

- *Regulatory policies*
- *Hardware limitations (for example, signal timing requirements)*
- *Interfaces to other applications*
- *Parallel operation*
- *Audit functions*
- *Control functions*
- *Higher-order language requirements*
- *Reliability requirements*

- *Criticality of the application*
- *Safety and security considerations*
- *Usability (including 508 compliance)*

*This section of the SDD should contain all the software design to a level of detail sufficient to enable programmers to develop a system that satisfies the requirements defined in the RSD. It should be detailed so as to make it easy for technical staff to find the methods to complete the designed function.*

*These requirements should, at minimum, include the following items:*

- *An indication of the associated requirement(s) in the RSD which is being designed*
- *A description of the functionality being designed*
- *The design entities (and their attributes) affected*
- *The algorithm executed (where appropriate) to implement the functionality.*

*Because the Dependencies and Constraints section is often the largest and most important part of the SDD, the following principles apply:*

- *Specific design should be cross-referenced to earlier, related documents (e.g., the RSD).*
- *All design should be uniquely identifiable.*
- *Items in this section should be identified from a technical level rather than an end user level. (i.e., an option name should be identified rather than the menu text for that option).*

## **6.2.2. Specific Requirements**

### **6.2.2.1. Database Repository**

*The Database Repository section in the RSD can be referenced in this section.*

*If a logical database design is a part of the system, it should be listed here. Logical database design should specify the logical requirements for any information that is to be placed into a database. This may include:*

- *Types of information used by various functions*
- *Frequency of use*
- *Accessing capabilities*
- *Data entities and their relationships*
- *Integrity constraints*
- *Data retention requirements.*

*Recommendation: Create a block diagram showing the databases and where the data resides.*

#### 6.2.2.2. System Features

*Describe the system features, functional requirements, sub-requirements, etc. which can be organized in an outline format that matches the RSD. Specific formatting and organization of the paragraphs (i.e., section numbering) is left to the discretion of the author and is dependent on the level of detail essential to fully describe the design. Some designs may only require two levels; others may require multiple levels. The information necessary to define the items or to specify modifications to the items affected by the functionality being designed should be provided in the appropriate design element tables. Where feasible, instead of duplicating the RSD, it can be referenced via a link, to avoid unnecessary duplication. The key goal is to provide traceability to requirements.*

#### 6.2.2.3. Design Element Tables

*The design element tables are provided for your convenience. Copy each table as many times as necessary to address multiple items within each section. Add rows and headings to the tables to provide any additional required information to define the item or to specify the modifications to the item. Numbering of the design element tables to align them underneath the applicable requirement or sub-requirement is recommended, but is left to the author's discretion. For that reason they are not numbered in this template.*

##### 6.2.2.3.1. Routines (Entry Points)

*This section is an illustration that is Vista specific. The authors are free to organize this information by technology, different templates, or optional sections depending on the task at hand.*

*Complete the table for each routine affected by the functionality being designed.*

**Table 23: Routines (Instructions)**

Routines	Instructions
Routine Name	<i>List the routine affected by the functionality being designed.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
RTM	<i>List the RSD item number within the SDD (i.e., If the RSD has a requirement of 3.3.1, add Support for a new API, then in this column list RSD Requirement 3.3.1)</i>
Related Options	<i>List options that directly call or are called by the routine.</i>
Related Routines	<i>List routines that directly call or are called by the routine.</i>
Data Dictionary (DD) References	<i>List files that reference the routine through input transforms, cross reference logic, etc.</i>
Related Protocols	<i>List protocols that reference or are referenced by the routine.</i>
Related Integration Control Registrations (ICRs)	<i>List proposed new ICRs and subscribed ICRs. Also, list any obscure Supported ICRs.</i>

Routines	Instructions
Data Passing	<i>Check the appropriate box. Also a short description of what invokes the new/changed routine should be included in this section. An example of such a description would be a note that the new/changed routine will be invoked as part of a function call or it would be invoked through user menu-driven options, system protocols, HL7 Logical Links, etc. This section refers specifically to the change implemented with the design.</i>
Input Attribute Name and Definition	<i>List the Input Attributes passed into the new or changed routine logic. Each attribute should be defined.</i>
Output Attribute Name and Definition	<i>List the Output Attributes returned from the new or changed routine logic. Each attribute should be defined.</i>
Current Logic	<i>Define the current logic in the routine that the design will modify. If this is new code, enter "N/A".</i>
Modified Logic (Changes are in bold)	<i>Define the logic in the routine that the design will implement.</i>

**Table 24: (Grouping): Routines**

Routines	Activities
Routine Name	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
RTM	
Related Options	

Related Routines	Routines "Called By"	Routines "Called"

Routines	Activities
Data Dictionary (DD) References	
Related Protocols	
Related Integration Control Registrations (ICRs)	
Data Passing	<input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local
Input Attribute Name and Definition	Name: Definition:

Routines	Activities
Output Attribute Name and Definition	Name: Definition:

Current Logic

Modified Logic (Changes are in bold)

#### 6.2.2.3.2. Templates

*Complete Table 25 for each template affected by the functionality being designed. A short description of what change will be made to the templates should be included in this section.*

*Note: If preferred, copy and paste this section directly from VA FileMan DDs instead of using the tables.*

**Table 25: Templates (Instructions)**

Templates	Instructions
Template Name	<i>Identify the template affected by the functionality being designed</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
RSD Traceability	<i>List the Requirement Specification Document (RSD) item number within the SDD (i.e., If the RSD has a requirement of 3.3.1, add Support for a new API, then this column should list RSD Requirement 3.3.1)</i>
Template Type	<i>Indicate the type of template identified (Sort, Input, or Print).</i>
Related Options	<i>List options that directly call or are called by the template.</i>
Related Routines	<i>List routines that directly call or are called by the template.</i>
Data Dictionary (DD) References	<i>List files/fields that reference the template(s) through input transforms, and cross reference logic.</i>
Global References	<i>List the ICRs for global references that are outside your namespace.</i>

**Table 26: Templates**

Templates	Description
Template Name	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
RSD	
Template Type	<input type="checkbox"/> Sort <input type="checkbox"/> Input <input type="checkbox"/> Print <input type="checkbox"/> Other

Templates	Description
Template Name	
Related Options	

Related Routines	Routines “Called By”	Routines “Called”

Routines	Description
Data Dictionary (DD) References	
Global References	

### 6.2.2.3.3. Bulletins

*If the project develops or affects bulletins, then complete this section; if not then state that the section is not applicable and delete the tables and content of the section. Complete the table for each bulletin affected by the functionality being designed. A short description of what change will be made to the bulletins should be included in this section.*

*Note: If preferred, copy and paste this section directly from VA FileMan DDs instead of using the tables.*

**Table 27: Bulletins (Instructions)**

Bulletins	Instructions
Bulletin Name	<i>List the specific bulletin affected by the functionality being designed.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
RTM	<i>List the RSD item number within the SDD (i.e., If the RSD has a requirement of 3.3.1, add Support for a new API, then in this column list RSD Requirement 3.3.1).</i>
Related Options	<i>List options that directly send the bulletin.</i>
Related Routines	<i>List routines that directly send the bulletin.</i>
Mail Subject	<i>List the subject of the mail message, i.e., which bulletin this affects.</i>
Mail Group	<i>List the mail group (recipients) of the mail message.</i>
Parameters	<i>List necessary parameters.</i>
Data Dictionary (DD) References	<i>List files/fields that reference the bulletin(s) through input transforms, cross reference logic, etc. should be listed under Data Dictionary (DD) References.</i>



**Table 28: Bulletins**

Bulletins	Description
Bulletin Name	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
RTM	

Related Routines	Routines "Called By"	Routines "Called"

Routines	Description
Mail Subject	
Mail Group	
Parameters	
Data Dictionary (DD) References	

#### **6.2.2.3.4. Data Entries Affected by the Design**

*Provide the following data for each field to be created, modified, or deleted or provide a "Before and After: Data Entries Affected by the Design."*

*Identify the entries affected by the design. If a blanket change will be made to each entry affected, that change should be defined in this table.*

*Only changes that are unique to each record should be defined in the Unique Record(s) section (Section 6.2.2.3.5). Redundant information should not be entered into each chart in the Unique Record(s) section.*

**Table 29: Data Entries Affected by the Design**

Field Name	Current Value	New Value

#### **6.2.2.3.5. Unique Record(s)**

*List the unique record ID(s) that will be affected by the changes implemented by the design. This is commonly done in the .01 field. The values defined in the Current Value and New Value columns should be the exact value of the data. For each unique record ID, copy this table and provide the information.*

**Table 30: Unique Record ID**

Field Name(s)	Current Value	New Value

#### 6.2.2.3.6. File or Global Size Changes

*Indicate the change to the size of the file or global as a result of the design implemented with this description. Global size changes tie back to the business requirements and RSD. Growth or reduction in the size of the global should be indicated in this section. If the file is static across all VistA systems, a blanket statement of how the change will affect the size of the global will suffice.*

*For example, “The National Procedure file is a new file and will require 8.7K of disk space to install.”*

*If a file is dynamic and its size may vary from VistA system to VistA system, the description should indicate the change in the file per record and the number of records that the site may anticipate. For example, if a field is being added to the patient file that will result in an increase of 7K per patient, the site can estimate the global growth based on the number of entries in that file.*

*Note: If the Capacity Planning analysis is available, then enter it here. If not, then use the Project Team projection.*

**Table 31: File or Global Size Changes**

File/Global Name(s)	Estimated Increase	Estimated Decrease

#### 6.2.2.3.7. Mail Groups

*Complete the table for each of the mail groups affected by the functionality being designed. A short description of what changes will be made to the affected mail groups should be included in this section.*

*Note: If preferred, this can be captured directly from VA FileMan DDs after the fact.*

**Table 32: Mail Groups (Instructions)**

Mail Groups	Instructions
Mail Group Name	<i>List the name of the mail group being modified. The mail group name may include a domain name.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
Related Options	<i>List options that directly reference the file.</i>
Related Routines	<i>List routines that reference the mail group.</i>
Data Dictionary (DDs) References	<i>List files that reference the mail group through input transforms, cross-reference logic, etc.</i>
Related Protocols	<i>List protocols that directly reference the mail group.</i>
Mail Group Description	<i>Describe the purpose for the mail group.</i>
Self-Enrollment Allowed	<i>Check the appropriate box either Yes or No.</i>
Type	<i>Check the appropriate box either Public or Private.</i>



**Table 33: Mail Groups**

Mail Groups	Activities
Mail Group Name	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
Related Options	

Related Routines	Routines “Called By”	Routines “Called”

Mail Groups	Instructions
Data Dictionary (DD) References	
Related Protocols	
Mail Group Description	
Self-Enrollment Allowed	<input type="checkbox"/> Yes <input type="checkbox"/> No
Type	<input type="checkbox"/> Public <input type="checkbox"/> Private

#### 6.2.2.3.8. Security Keys

*This section lists the specific security keys affected by the functionality being designed. A short description of the changes that will be made to the security keys affected should be included in this section.*

*Note: If preferred, this can be captured directly from VA FileMan DDs after the fact.*

**Table 34: Security Keys (Instructions)**

Security Keys	Instructions
Security Key Name	<i>List the specific name of the security key being modified.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
Related Options	<i>List options that directly reference the security key.</i>
Related Routines	<i>List routines that reference the security key.</i>
Data Passing	<i>Check the appropriate box. Enter a short description of an event that would trigger the new/changed routine, for example, a note that the change to the security key will be referenced through user menu driven options, routines, etc. This section refers specifically to the change implemented with the design.</i>
Security Key Description	<i>List a brief description of the security key.</i>
Subordinate Keys	<i>List any subordinate keys.</i>

Security Keys	Instructions
<b>Mutually Exclusive Keys</b>	<i>Enter the name of a key that may not be held jointly with this one.</i>
<b>Granting Condition Logic</b>	<i>Define the logic for the Granting Condition of the Security Key affected by the functionality being designed.</i>
<b>Current Logic</b>	<i>If the security key currently has a granting condition, define the current logic for that granting condition. If the security key did not exist before, indicate that there is currently no security key.</i>
<b>Modified Logic (Changes are in bold)</b>	<i>Define the granting condition that the design will implement. If the security key is new to the field, define the logic here.</i>
<b>Hierarchical Precedence</b>	<i>Define which key is used if one key will take precedence over another key.</i>

**Table 35: Security Keys**

Security Keys	Activities
<b>Security Key Name</b>	
<b>Enhancement Category</b>	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
<b>Related Options</b>	

Related Routines	Routines “Called By”	Routines “Called”

Security Keys	Activities
<b>Data Passing</b>	<input type="checkbox"/> Input <input type="checkbox"/> Output <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local Reference
<b>Security Key Description</b>	
<b>Subordinate Keys</b>	
<b>Mutually Exclusive Keys</b>	
<b>Granting Condition Logic</b>	

Current Logic

Modified Logic (Changes are in bold)

Security Keys	Activities
Hierarchical Precedence	

#### 6.2.2.3.9. Options

*Complete the table for each of the options affected by the functionality being designed. A short description of the changes that will be made to the options affected should be included. Changes to the OPTION file (#19) are to be included, not the functionality of the option invoked.*

*Note: If preferred, this can be captured directly from VA FileMan DD after the fact.*

**Table 36: Options (Instructions)**

Options	Instructions
<b>Option Name</b> (MENU TEXT field)	<i>Enter the name of the option affected.</i>
<b>Enhancement Category</b>	<i>Check the appropriate box: New, Modify, Delete, or No Change</i>
<b>Associated Menu</b> <b>Options that will invoke</b> <b>this reference</b>	<i>List the menu type options on which the respective option is or will be contained.</i>
<b>Data Passing</b>	<i>Check the appropriate box. Also a short description of what invokes the new/changed routine should be included in this section. An example of such a description would be a note that the change to the option will be referenced through VA Mailman server messages, user selection of the option from the VA Kernel Menu Management system, etc. This section refers specifically to the change implemented with the design.</i>
<b>Menu Text Description</b>	<i>Enter the name of the option as it will be displayed to the user within the menu system.</i>
<b>Option Type</b>	<i>Specify the type of option</i>
<b>Option Definition</b>	<i>Provide all the information necessary to fully define the option. Include options that are included in the menu, if applicable.</i>
<b>Current Entry Action</b> <b>Logic</b>	<i>Define the current logic for the entry action of the option affected by the functionality being designed. If the entry action did not exist before, indicate that there currently is no entry action.</i>
<b>Modified Entry Action</b> <b>Logic (Changes are in</b> <b>bold)</b>	<i>Define the entry action that the design will implement. If the entry action is new to the field, define the logic here.</i>
<b>Current Exit Action</b> <b>Logic</b>	<i>Define the current logic for the exit action of the option affected by the functionality being designed. If the exit action did not exist before, indicate that there currently is no exit action.</i>
<b>Modified Exit Action</b> <b>Logic</b> <b>(Changes are in bold)</b>	<i>Define the exit action that the design will implement. If the exit action is new to the field, define the logic here.</i>

**Table 37: Options**

Options	Activities
Option Name	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
Associated Menu Options that will invoke this reference	
Data Passing	<input type="checkbox"/> Input <input type="checkbox"/> Output <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local Reference
Menu Text Description	
Option Type	<input type="checkbox"/> Edit <input type="checkbox"/> Print <input type="checkbox"/> Menu <input type="checkbox"/> Inquire <input type="checkbox"/> Action <input type="checkbox"/> Run Routine <input type="checkbox"/> Other
Associated Routine	
Option Definition	

Current Entry Action Logic

Modified Entry Action Logic (Changes are in bold)

Current Exit Action Logic

Modified Exit Action Logic (Changes are in bold)

#### 6.2.2.3.10. Protocols

Complete the table for each of the protocols affected by the functionality being designed. A short description of the changes that will be made to the protocols affected should be included in this section. Changes to the PROTOCOL file (#101) are to be included, not the functionality of the protocol invoked.

Note: If preferred, this can be captured directly from VA FileMan DDs after the fact.

**Table 38: Protocols (Instructions)**

Protocols	Instructions
<b>Protocol Name</b>	List the name of the protocol affected.
<b>Enhancement Category</b>	Check the appropriate box: New, Modify, Delete, or No Change.
<b>Associated Protocols</b>	List the ancestors of the protocol being designed, i.e., those protocols that contain the respective protocol as an item.
<b>Data Passing</b>	Check the appropriate box. An event that would trigger the new/changed protocol should be included in this section. An example would be a note that the change to the protocol will be referenced through the VA event driver, List Manager, user selection of a protocol from the VA Kernel Menu Management system. This section refers specifically to the change implemented with the design.
<b>Item Text Description</b>	Enter the protocol's text as it appears to the user on the menu or sub-header.
<b>Protocol Type</b>	Define the type of protocol to be executed
<b>Associated Routine</b>	List any associated routines affected by the protocol being designed.
<b>Current Entry Action Logic</b>	Define the current logic for the entry action of the protocol affected by the functionality being designed. If the entry action did not exist before, indicate that there currently is no entry action.
<b>Modified Entry Action Logic (Changes are in bold)</b>	Define the entry action that the design will implement. If the entry action is new to the field, define the logic here.
<b>Current Exit Action Logic</b>	Define the current logic for the exit action of the protocol affected by the functionality being designed. If the exit action did not exist before, indicate that there currently is no exit action.
<b>Modified Exit Action Logic (Changes are in bold)</b>	Define the exit action that the design will implement. If the exit action is new to the field, define the logic here.

**Table 39: Protocols**

Protocols	Activities
Protocol Name	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
Associated Protocols	
Data Passing	<input type="checkbox"/> Input <input type="checkbox"/> Output <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local Reference
Item Text Description	
Protocol Type	<input type="checkbox"/> Action <input type="checkbox"/> Menu <input type="checkbox"/> Protocol <input type="checkbox"/> Protocol Menu <input type="checkbox"/> Limited Protocol <input type="checkbox"/> Extended Action <input type="checkbox"/> Dialog <input type="checkbox"/> Other
Associated Routine	

Current Entry Action Logic

Modified Entry Action Logic (Changes are in bold)

Current Exit Action Logic

Modified Exit Action Logic (Changes are in bold)

#### 6.2.2.3.11. Remote Procedure Call (RPC)

*Complete the table for each RPC affected by the functionality being designed.*

*Note: If preferred, this can be captured directly from VA FileMan DDs after the fact.*

**Table 40: RPCs (Instructions)**

RPCs	Instructions
Name	<i>List the specific name of the RPC affected.</i>
TAG^RTN	<i>List the tag (label) and routine.</i>
Input Parameters	<i>This field is used to identify an input parameter for the API.</i>
Results Array	<i>This field tells the RPC Broker how to process the resulting data from the call.</i>
Description	<i>Provide a brief description of the RPC affected.</i>

**Table 41: RPCs**

RPCs	Activities
Name	
TAG^RTN	
Input Parameters	
Results Array	<input type="checkbox"/> Single Value <input type="checkbox"/> Array <input type="checkbox"/> Word Processing <input type="checkbox"/> Global Array <input type="checkbox"/> Global Instance
Description	

**6.2.2.3.12. Constants Defined in Interface**

*Provide the name and description.*

**Table 42: Constants Defined in Interface**

Name	Description

**6.2.2.3.13. Variables Defined in Interface**

*Provide the name, type, and description.*

**Table 43: Variables Defined in Interface**

Name	Type	Description

**6.2.2.3.14. Types Defined in Interface**

*Provide the name, type, and description.*

**Table 44: Types Defined in Interface**

Name	Type	Description

**6.2.2.3.15. GUI**

*List the GUI affected by the functionality being designed and include a short description of the changes made to the affected GUI. The headers in the following tables have names for the information outlined. There are a number of items in this section that would generally be global information and visible to all other aspects.*

**Table 45: GUI**

Unit Name	Description



#### 6.2.2.3.16. GUI Classes

**Table 46: GUI Classes (Instructions)**

GUI Classes	Instructions
Class Name	<i>List the name of the class affected. The headers in the following tables have names for the information outlined. Note that only the new properties and methods for a class are listed below. All ancestor properties and methods are still available and unchanged.</i>
Derived From Class	<i>List the class that this is derived from, its parent and any interfaces listed as part of this class.</i>
Purpose	<i>Describe the functionality that users can access from this class and related form, if any.</i>

**Table 47: GUI Classes**

GUI Classes	Instructions
Class Name	
Derived From Class	
Purpose	

#### 6.2.2.3.17. Current Form

*Provide a screen capture or graphical representation of the current layout.*

#### 6.2.2.3.18. Modified Form

*Provide a screen capture or graphical representation of the layout that the design will implement.*

#### 6.2.2.3.19. Components on Form

**Table 48: Components on Form**

Name	Type	Description

#### 6.2.2.3.20. Events

**Table 49: Events**

Name	Type	Description

#### 6.2.2.3.21. Methods

**Table 50: Methods**

Method Name	Procedure/Function	Description

#### 6.2.2.3.22. Special References

*Include references that are not listed elsewhere.*

**Table 51: Special References**

Special Reference Name	Type	Description

#### 6.2.2.3.23. Class Events

**Table 52: Class Events**

Name	Type	Description

#### 6.2.2.3.24. Class Methods

**Table 53: Class Methods**

Name	Procedure/Function	Description

#### 6.2.2.3.25. Class Properties

**Table 54: Class Properties**

Class Properties Name	Type	Visibility	Description

#### 6.2.2.3.26. Uses Clause

*Use this section to provide a uses clause that lists the other units (code or form units) that this unit will use. This may be documented in the form of a Unified Modeling Language (UML) drawing.*

#### 6.2.2.3.27. Forms

*This section lists the forms that will be affected or created by the functionality being designed. A short description of the change that will be made to the forms should be included.*

**Table 55: Forms (Instructions)**

Forms	Instructions
Form Name	<i>List the name of the form affected by the functionality being designed.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
Form Functionality	<i>Describe the form's functionality and refer to the usage of the form. An example of such a description is "This form is used to enter patient demographic data."</i>
Current Form Layout	<i>Define the current form layout that the design will modify. If this is a new form, enter "N/A".</i>

Forms	Instructions
<b>Modified Form Layout</b> (Changes are in bold)	<i>Define the form layout that the design will implement.</i>

**Table 56: Forms**

Forms	Description
<b>Form Name</b>	
<b>Enhancement Category</b>	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
<b>Form Functionality</b>	

Current Form Layout

Modified Form Layout (Changes are in bold)

#### 6.2.2.3.28. Functions

*The functions affected by the capabilities being designed should be listed in this section. A short description of what change will be made to the functions and/or new functions should be included.*

**Table 57: Forms (Instructions)**

Functions	Instructions
<b>Function Name</b>	<i>List the specific function affected by the capability being designed.</i>
<b>Short Description</b>	<i>List a short description of the change that will be made to the functions and/or new functions.</i>
<b>Enhancement Category</b>	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
<b>Related Options</b>	<i>List the options that directly call or are called by the function.</i>
<b>Related Routines</b>	<i>List the routines that directly call or are called by the function.</i>
<b>Data Dictionary (DD) References</b>	<i>List the files that reference the function through input transforms, cross reference logic, etc.</i>
<b>Related Protocols</b>	<i>List the protocols that reference or are referenced by the function.</i>
<b>Related Integration Control Registrations (ICRs)</b>	<i>List proposed new ICRs and subscribed ICRs. Also, list any obscure Supported ICRs.</i>

Functions	Instructions
Data Passing	<i>Check the appropriate box. An event that would trigger the new/changed function should be included in this section. An example of such a description would be a note that the new/changed function will be invoked as part of a function call or it would be invoked through system protocols, HL7 Logical Links, etc. This section refers specifically to the change implemented with the design.</i>
Input Attribute Name and Definition	<i>List the input attributes passed into the new or changed function logic. Each attribute should be defined.</i>
Output Attribute Name and Definition	<i>List the output attributes returned from the new or changed function logic. Each attribute should be defined.</i>
Current Logic	<i>Define the current logic in the function that the design will modify. If this is new code, enter "N/A".</i>
Modified Logic (Changes are in bold)	<i>Define the logic in the function that the design will implement.</i>

**Table 58: Forms**

Function Name	Activities
Short Description	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
Related Options	

Related Routines	Routines "Called By"	Routines "Called"

Function Name	Activities
Data Dictionary (DD) References	
Related Protocols	
Related Integration Control Registrations (ICRs)	
Data Passing	<input type="checkbox"/> Input <input type="checkbox"/> Output <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local Reference

Function Name	Activities
Input Attribute Name and Definition	Name: Definition:
Output Attribute Name and Definition	Name: Definition:

Current Logic

Modified Logic (Changes are in bold)

#### 6.2.2.3.29. Dialog

*In this section list the changes to the DIALOG file (#.84).*

**Table 59: Dialog (Instructions)**

Dialog	Instructions
Dialog Message (Description)	<i>List the specific message affected or needed by the changes being designed.</i>
Enhancement Category	<i>Select the appropriate category: New, Modify, Delete, or No Change.</i>
Dialog Message (Description) Condition	<i>Describe the dialog message (description) functionality. An example of such a description would be the condition that would trigger the output of the message (dialog). This section refers to the condition generating the message (dialog).</i>
Current Dialog Message (Description)	<i>Define the current dialog message (description) that the design will modify. If this is a new dialog message (description) enter N/A.</i>
Modified Dialog Message (Description) (Changes are in bold)	<i>Define the dialog message (description) that the design will implement.</i>

**Table 60: Dialog**

Dialog	Instructions
Dialog Message (Description)	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
Dialog Message (Description) Condition	

Dialog	Instructions
Current Dialog Message (Description)	
Modified Dialog Message (Description) (Changes are in bold)	

#### 6.2.2.3.30. Help Frame

*A short description of what change will be made to the Help Frame text and/or new text should be included in this section. Help frames may be associated with options or with data dictionary fields to provide on-line instruction.*

**Table 61: Help Frame (Instructions)**

Help Frame	Instructions
Help Frame Text	<i>List the text affected or needed by the changes being designed.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
Help Frame Text Calling Mechanism	<i>Provide a short description of the mechanism used to call the Help Frame text in this section. An example of a mechanism would be the name of the routine or an explanation of how the Help Frame is called. An example of a calling mechanism would be the Standard VA FileMan API and the keystroke(s) that would trigger the output of the text.</i>
Current Help Frame Text	<i>List the current Help Frame Text that the design will modify. If new text enter N/A.</i>
Modified Help Frame Text (Changes are in bold)	<i>List the Help Frame Text that the design will modify.</i>

**Table 62: Help Frame**

Help Frame	Description
Help Frame Text	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change
Help Frame Text Calling Mechanism	

Current Help Frame Text

Modified Help Frame Text (Changes are in bold)

#### 6.2.2.3.31. HL7 Application Parameter

**Table 63: HL7 Application Parameter (Instructions)**

HL7 Application Parameter	Instructions
HL7 Application Parameter Name	<i>List the HL7 Application Parameter affected or needed by the changes being designed.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
Application Status	<i>Check the appropriate box in the applicable column for Current and Modified</i>
Facility Name	<i>List the current and modified value in the appropriate column.</i>
Country Code	<i>List the current and modified value in the appropriate column.</i>
HL7 Field Separator	<i>List the current and modified value in the appropriate column.</i>
HL7 Encoding Characters	<i>List the current and modified value in the appropriate column.</i>
Mail Group	<i>List the current and modified value in the appropriate column.</i>

**Table 64: HL7 Application Parameter**

HL7 Application Parameter Name	Description	
Enhancement Category	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change	
Application Status	<input type="checkbox"/> Active <input type="checkbox"/> Inactive <input type="checkbox"/> Active <input type="checkbox"/> Inactive	
Enhancement Category	<b>Current</b>	<b>Modified</b>
Facility Name		
Country Code		
HL7 Field Separator		
HL7 Encoding Characters		
Mail Group		

#### 6.2.2.3.32. HL7 Logical Link

**Table 65: HL7 Logical Link (Instructions)**

HL7 Logical Link	Instructions
HL7 Logical Link Parameter (LLP) Name	<i>List the specific HL7 Logical Link affected or needed by the changes being designed.</i>
Enhancement Category	<i>Check the appropriate box: New, Modify, Delete, or No Change.</i>
Node	<i>List the current and modified value in the appropriate column.</i>
Institution	<i>List the current and modified value in the appropriate column.</i>
Domain	<i>List the current and modified value in the appropriate column.</i>
Autostart	<i>List the current and modified value in the appropriate column.</i>
Queue Size	<i>List the current and modified value in the appropriate column.</i>



<b>HL7 Logical Link</b>	<b>Instructions</b>
<b>LLP Type</b>	<i>List the current and modified value in the appropriate column.</i>

**Table 66: HL7 Logical Link**

<b>HL7 Logical Link</b>	<b>Description</b>	
<b>HL7 Logical Link Parameter Name</b>		
<b>Enhancement Category</b>	<input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change	
<b>Enhancement Category</b>	<b>Current</b>	<b>Modified</b>
<b>Node</b>		
<b>Institution</b>		
<b>Domain</b>		
<b>Autostart</b>		
<b>Queue Size</b>		
<b>LLP Type</b>		

#### **6.2.2.3.33. COTS Interface**

*The specific communication method(s) and Application Interface(s) that will be created or modified for the COTS system being interfaced should be described in this section. A short description of the existing tools that will be used and any new tools that will be developed should also be included.*

**Table 67: COTS Interface (Instructions)**

<b>COTS Interface</b>	<b>Instructions</b>
<b>Communication Method</b>	<i>List the specific communication method created or modified for the functionality being designed.</i>
<b>Application Interface</b>	<i>List the specific application interface created or modified for the functionality being designed.</i>

**Table 68: COTS Interface**

<b>COTS Interface</b>	<b>Description</b>
<b>Communication Method</b>	
<b>Application Interface</b>	

## 6.3. Network Detailed Design

*Provide enough detailed information about the communication requirements to build and/or procure the communication components for the system. This section should provide sufficient detail to support the procurement of hardware for the system installation. Include the following information in the form of detailed designs (as appropriate):*

- *Details of servers and clients to be included on each area network*
- *Specifications for bus timing requirements and bus control*
- *Format(s) for data being exchanged between components*
- *Diagrams showing connectivity between components, data flow (if applicable), and distances between components*
- *LAN topology.*

## 6.4. Security and Privacy

### 6.4.1. Security

In accordance with the Federal Information Processing Standard 199 (FIPS 199) analysis, the Security Categorization for the Beneficiary Travel Self-Service System (BTSSS) is ***HIGH***. The Security Categorization will drive the initial set of minimal security controls required for the information system. Minimum security control requirements are addressed in NIST SP 800-53 and VA Handbook 6500, Appendix F. All VA security requirements, as defined in VA Handbook 6500 Appendix F, will be adhered to.

VA Common Security Controls are documented in [VA Handbook 6500 Appendix F, attachment-1](#). Common Controls are applicable to all OI&T systems and are the responsibility of OI&T management for implementation and management.

VA Hybrid Security Controls are documented in VA Handbook 6500 Appendix F, attachment-2. Hybrid Security Controls can be defined as part common control and part system-specific control. The implementation of these controls is the responsibility of the field, either at the national, regional, or facility system level.

VA System-Specific Security Controls are documented in VA Handbook 6500 Appendix F, attachment-3. Applicable system-specific controls should be determined by the Information System Owner, using the recommended values provided in VA Handbook 6500 Appendix F, attachment-3. System-specific controls may be tailored to meet the unique specifications and environment of the system, as determined by the Information System Owner.

The BTSSS Requirement Specification Document (RSD), section 2.13, documents the Security Specifications, as well as business specific security requirements.

### 6.4.2. Privacy

Protecting the privacy of data that BTSSS will be managing, whether it is transactional, unstructured, or meta-data, is of utmost importance to the system design and functionality.

There are both privacy and data security constraints that should be addressed in accordance with VA directives and Health Insurance Portability and Accountability Act (HIPAA) Privacy Act.

The BTSSS will adhere to all proposed VA Privacy requirements and controls, Identity Management and Security requirements, including VA Handbook 6500, March 2015, Appendix C: (References), Appendix E: (VA System Privacy Controls), and NIST SP 800-53 Revision 4 Privacy Controls. Efforts that involve the collection and maintenance of Personal Identifiable Information (PII) must be covered by a Privacy Act system of records notice. The publication [1605 Notice of Privacy Practices](#) can be obtained by clicking on the hyperlink.

The VHA Health Care Security Requirements (HCSR) will determine applicable HIPAA security requirements for the BTSSS project.

## **6.5. Service Oriented Architecture / ESS Detailed Design**

*This section provides details of provided and consumed services as follows:*

- *Consumed Services: Provide link to Service Description Document for each consumed service.*
- *Provided Services: Give service design for each provided service.*

*The information you provide here will be used to upload to the ESS Registry and Repository. At some point in the near future, we do not expect these SOA artifacts such as SLA, Service Description, etc. to be static documents. They will be dynamically generated from the ESS Registry and Repository tool in the form of reports. Any application and service integration design is also documented here.*

*A list of currently available Enterprise Shared Services is available here: <insert link to ESS list>*

### **6.5.1. Service Description for <Consumed Service Name>**

*Provide link to Service Description document for the consumed service. This section will repeat for each consumed service. The Service Description includes Service Interface and Service Level Definition (SLD) to address anticipated capacity requirements.*

### **6.5.2. Service Design for <Provided Service Name>**

*This section should describe the detailed service design for each ESS and SOA service needed to obtain an intended result. The Service Design includes Service Interface and Service Level Definition (SLD) to address anticipated capacity requirements.*

*This section will repeat for each provided service.*

#### **6.5.2.1. Introduction**

##### **6.5.2.1.1. Purpose and Scope of Service**

*This service was described at a high level in the charter document. Please refer to it here via a link.*

#### 6.5.2.1.2. Links to Other Documents

*Provide links to other documents created for this service so far in the SOA lifecycle. At a minimum, provide links to:*

- *Service Charter*
- *Service Roadmap*
- *Service Description*

#### 6.5.2.2. Service Details

##### 6.5.2.2.1. Service Identification

*This section will be written as a table to provide a quick reference to the service's what, where, why and how - cheat sheet.*

**Table 69: Service Identification**

Service Attribute	Value
<i>Name and Alias (if any)</i>	<i>Name of the service and other names for the service, which might be used by someone searching for this service. Please follow ESS naming standards.</i>
<i>Overview</i>	<i>Brief textual overview of the service.</i>
<i>Version</i>	<i>Version number of the service being described here</i>
<i>Latest Status</i>	<i>This field shows the latest status for the above referenced version of this service! The status of a service shows the progress of the service from initiation through development, deployment, and eventual retirement. The status also has a status date associated with the status - and we will be using the latest one here in this document. Valid values include: Inception, Design, Provisioning, Certification / Testing, Operation, Deprecated, Retired, Rejected - Owner has decided not to develop the service.</i>
<i>Service Type</i>	<i>Used to define applicable architecture patterns. Examples (from Open Group):</i> <ul style="list-style-type: none"><li>• <i>Interaction</i></li><li>• <i>Process</i></li><li>• <i>Information</i></li><li>• <i>Partner</i></li><li>• <i>Business Application</i></li><li>• <i>Access</i></li><li>• <i>Service Connectivity</i></li></ul>

Service Attribute	Value
Architecture Layer	<i>Referred to as class in VA Service template. Used to define applicable architecture patterns and relationships to governing bodies. Examples:</i> <ul style="list-style-type: none"> <li>• <i>Solution</i></li> <li>• <i>Process</i></li> <li>• <i>Information</i></li> <li>• <i>Utility</i></li> <li>• <i>Underlying</i></li> </ul>
Business Domain	<i>Business Vertical or Business Division where this service belongs.</i>
Service Domain	<i>The service or technical domain that the service belongs to. Can be used to establish the namespace.</i>
Business Organization and Owner	<i>Person who approves this service &amp; any changes. Include email.</i>
Technical Organization and Owner	<i>Person responsible for provisioning (specifying, acquiring certifying) this service. Include email.</i>
Development Organization and Owner	<i>Person who is responsible for the development processes and activities for this service. Include email.</i>
Support Organization and Owner	<i>Person who is responsible for the support of this service while in production. Include email.</i>
Target Consumer Organization(s) and Owner(s)	<i>Organizations and/or developers roles that service is intended for.</i>

#### 6.5.2.2.2. Service Versions

**Table 70: Service Versions**

Version Numbers	Current Status of Version	A Brief Description of the change implemented in that version
<i>This version</i>	<i>Being Designed</i>	
<i>Example: version 2</i>	<i>Example: In production. Will be retired with this release.</i>	<i>Example: This release added the ability to look up a person by address. Provide a link to each version of the service.</i>
<i>Example: version 1</i>	<i>Example: Retired.</i>	<i>Example: This release provided the base minimum functionality to look up a person by name. Provide a link to each version of the service.</i>

#### 6.5.2.2.3. Summary of Design and Platform Details

##### 6.5.2.2.3.1. SOA Pattern(s) Implemented

*Name of the SOA pattern implemented – for instance, this may be a Pub/Sub model. Just a name and reference to the document or book with the pattern is sufficient for popular patterns or VA's own patterns. If you are using some esoteric pattern, more details will help.*

##### 6.5.2.2.3.2. COTS Platform Vendor Names and Versions for Hosting Platform

*Example, TIBCO.*

### **6.5.2.3. Dependencies**

*The Dependency Model identifies other services, systems, databases, etc. that [Service Name] is dependent upon or interacts with to perform its function.*

*This section should clearly identify all sources and external systems that are accessed by this service to fulfill the service consumers' request. This section should include diagrams to show as much detail as necessary to inform the developer. Provide a context diagram for the service.*

*Note: Here our primary audience includes the providers of the service. So this document in general will emphasize system components and sub-systems as much as external interactions.*

### **6.5.2.4. Service Design Details**

*The next sub-section on Interface Technical Specs could be just a copy from the corresponding sub-section in Interface section in the Service Description Document. Here, you could provide more detail necessary for building this service but the interface spec needs to be consistent between this document and the Service Description Document. This section contains all information necessary to fully describe an interface published by this service...*

#### **6.5.2.4.1. Interface Technical Specs**

*The technical specification allows developers of service consumers to locate and discover the service for run time consumption.*

##### **6.5.2.4.1.1. Service Invocation Type**

*Such as: SOAP over HTTP, REST.*

##### **6.5.2.4.1.2. Service Interface Type**

*Such as: WSDL via Web Service 2.0*

##### **6.5.2.4.1.3. Service Name**

*Technical Service Name. Comply with ESS naming standards.*

##### **6.5.2.4.1.4. Interface**

*Link to WSDL or other interface document.*

##### **6.5.2.4.1.5. End Points**

*Provide if known! Calls that can be made into the service. Can be referenced to the WSDL or can be in a separate table.*

##### **6.5.2.4.1.6. Operations or Methods**

*In the table below, the technical names of the operations, inputs and outputs are used. Inputs and outputs, if parameters, must have a data type.*

*Non-primitive data types must be defined in the Service Information Model section.*

*This table could be generated automatically from the WSDL content or its equivalent.*

*Style can take any of these values: Parameters or Document; and One-way or Request-response or Solicit-response or Notification.*

*Use a separate column for the operation purpose if you wish.*



*You might use abbreviations in the Faults column and explain the abbreviations used below the table. For example, NF = Not Found, MI = Missing Input.*

**Table 71: Operations or Methods**

Operation Name	Inputs	Outputs	Transactional Qualities if relevant (Updating?, Atomic?, Can participate in transaction?)	Pre and Post Conditions	Exception (s)

*Provide a link to the Service Information model so that the consumer of your system knows the schema for the input and output parameters.*

#### **6.5.2.4.1.7. Message Schemas**

*Provide definitions or links to definitions of the message(s) related to the service operations. These may be dependent on the implementation style and protocol binding of the interface.*

#### **6.5.2.4.2. Information Model**

*Even though this section looks similar to the corresponding section 3.2 in Service Description, remember that the primary objective here is to facilitate construction and to gain approvals from governing bodies. So you will provide more of a “white box” view of the design here to help your developers code the service.*

##### **6.5.2.4.2.1. Class Diagram and Description of Entities Involved**

*Map out all entities involved in the service: input, output, exceptions, entities manipulated in persistent media/DBs, intermediate entities created in memory etc.*

##### **6.5.2.4.2.2. Mappings from ELDM to Standards Based Schemas**

*Provide mappings from your native schema to any standards based schemas your service will use to communicate outside. For instance, if you are using HL7 based messages then you will show how data is converted from your native schema to HL7.*

##### **6.5.2.4.3. Behavior Model (AKA Use Case Realization)**

*The Behavior Model defines the actions and processes supported by the service. Actions and methods represented in the use cases and sequence diagrams shown below are further defined by the operation contracts and the message payloads.*

##### **6.5.2.4.3.1. Use Cases (Use Case Model)**

*Describe how this service fits into the larger use case model of the consumer. You may need multiple models for multiple consumers. Focus is **not** on the internal workings of the new service instead of the calls made from external consumers. Just a summary or the Use Case Diagram may be sufficient. List the alternative and exception flows. Reference the detailed design documents via a URL.*

##### **6.5.2.4.3.2. Interaction Diagrams**



*Cut and paste screen shot from RSA or similar tool or provide link to the model. Provide description to help developers build your service. The interaction diagrams should depict external interactions and internal sequences of calls between internal components. The sequence diagram should cut through all layers to show the main, alternate and exception flows.*

#### **6.5.2.5. Gap Analysis**

*Provide a Gap Analysis (Reference) to demonstrate compliance of this service with various standards, policies, guidelines and laws. The Gap Analysis may take the form of a matrix as shown in the sample below. This will help the governance boards expedite your request.*

**Table 72: Gap Analysis**

<b>Design Elements→ Policies / SLD elements etc.↓</b>	<b>Design Element A</b>	<b>Design Element B</b>	<b>Design Element C</b>	<b>Comment for non-conformance</b>
<i>Policy X</i>	<i>Match</i>			
<i>Policy Y</i>		<i>Partial</i>		
<i>Policy Z</i>				<i>Commercial encryption server in prod will have to address this policy.</i>
<i>Policy A</i>				<i>Compliance with this policy not required until next year.</i>
<i>New / Additional Features</i>			<i>New element minimizes manual intervention</i>	

#### **6.5.2.5.1. Variances from Enterprise Target Architecture**

*This list of “variances” will become a submission to the ESS dispensation process.*

#### **6.5.2.5.2. Variances from SLDs**

*This list of “variances” will become a submission to the ESS dispensation process.*

#### **6.5.2.5.3. Variances from Standards and Policies**

*This list of “variances” will become a submission to the ESS dispensation process.*

#### **6.5.2.5.4. Justification for Exceptions and Mitigation**

*This section will list out any non-functional and functional requirements that are not being met. The non-conformance may be in violation of elements of SLDs, enterprise architecture (TRM Technology Reference Model), privacy policies or guidelines. For each exception provide:*

- 1. Reasons for non-conformance (cost, time, technology, etc.)*
- 2. Mitigating actions taken to reduce the impact of non-conformance*
- 3. Plan (roadmap) to come back into conformance*

*This list can grow depending on what the Review bodies may ask for.*

## 7. External System Interface Design

*This section details interfaces external to system, that are NOT services (ESS/SOA). Typically, these may include, RPCs, Flat Data Files etc.*

*External systems are systems that are not within the scope of the system under development, regardless of whether the other systems are managed by the vendor or its client.*

*In this section, describe the interface(s) between the system under development (i.e., the system that is the subject of this SDD) and external systems and/or subsystem(s).*

*It is best to illustrate these sections with annotated diagrams to clearly identify the various elements of the interfaces.*

### 7.1. Interface Architecture

*Describe the interface(s) between the system being designed and other systems. Include the interface architecture(s) being implemented, such as wide area networks, gateways, etc. Provide diagrams showing the communications path(s) between this system and other systems.*

### 7.2. Interface Detailed Design

*Provide sufficient detail about the interface requirements for the development team to format, transmit, and/or receive data across the interface.*

*Include the following information (as appropriate):*

- *Data format requirements; if data must be reformatted before it is transmitted or after incoming data is received. Describe the tools and/or methods for the reformat process.*
- *Specifications for hand-shaking protocols between systems; content and format of hand-shake messages, timing for exchanging these messages, and errors handling.*
- *Format(s) for reports exchanged between the systems.*
- *Graphical representation of the connectivity between systems, showing the direction of data flow.*
- *Query and response descriptions.*
- *Describe the individual data elements that the interfacing entity(s) will provide, store, send, access, and receive, such as:*
- *Names/identifiers*
  - *Data Element Name*
  - *Data Format/Length*
  - *Data Type*
  - *Definition*
  - *Non-Technical Name*
  - *Non-Technical Synonyms*
  - *Specifications*
  - *Synonyms*

- *Range or enumeration of possible values (e.g., 0-99)*
- *Accuracy and precision (number of significant digits)*
- *Priority, timing, frequency, sequencing, and other constraints*
- *Security and privacy constraints*
- *Sources (setting/sending entities) and recipients (using/receiving entities).*

*Describe the data element assemblies (records, messages, files etc.) that the interfacing entity(s) will provide, store, and send, such as:*

- *Names/identifiers*
  - *Technical Name, e.g., data structure name*
  - *Non-technical Names, e.g. synonyms*
- *Data elements*
- *Medium/structure of data elements/assemblies*
- *Visual characteristics (e.g. layouts, fonts, icons etc.)*
- *Relationships among assemblies*
- *Security and privacy constraints*
- *Sources and recipients.*

*Describe the communication methods that the interfacing entity(s) will use for the interface, such as:*

- *Communication links, bands, frequencies, and media*
- *Message formatting*
- *Flow control (e.g. sequence numbering)*
- *Data transfer rate*
- *Routing*
- *Transmission services*
- *Safety*
- *Security and privacy considerations.*

*Describe characteristics of the protocols that the interfacing entity(s) will use for the interface, such as:*

- *Priority/layer of the protocol*
- *Packeting*
- *Legality checks, error control*
- *Recovery procedures*
- *Synchronization*
- *Status, identification, and other reporting features.*

*Where appropriate describe other characteristics, such as physical compatibility of the interfacing entity(s) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)*

## 8. Human-Machine Interface

*Describe the human-machine interface (i.e., GUI) relative to the user. Additional information may be added if the suggested headings are inadequate.*

### 8.1. Interface Design Rules

*Identify conventions and standards for designing the GUI.*

### 8.2. Inputs

*Identify the input media used by the user (i.e., operator) for providing information to the system, such as data entry screens, optical character readers, bar scanners, etc.*

*Identify the messages associated with operator inputs, including the following:*

- *Form(s) if the input data is keyed or scanned for data entry*
- *Access restrictions*
- *Security considerations.*

### 8.3. Outputs

*Describe the system output design relative to the user. System outputs include reports, data display screens, query results, etc.*

*Identify the following, if appropriate:*

- *Access restrictions or security considerations*
- *Description of the purpose of the output*
- *Report requirements, including frequency of periodic reports*
- *Screen contents. (Provide a graphic representation of each layout. Define all data elements associated with the layout).*

### 8.4. Navigation Hierarchy

*Provide a diagram of the navigation hierarchy that shows how a user moves through the GUI.*

#### 8.4.1. Screen [x.1]

*Provide the layout of all input data screens or GUIs. Provide a graphic representation of each GUI, for example, a low-resolution screenshot. Define all data elements associated with each screen or GUI, or reference the data dictionary. Label each data input screen and/or GUI.*

#### 8.4.2. Screen [x.2]

*Provide a graphic representation of each GUI, for example, a low-resolution screenshot. Define all data elements associated with each screen or GUI, or reference the data dictionary.*

#### 8.4.3. Screen [x.3]

*Provide a graphic representation of each GUI, for example, a low-resolution screenshot. Define all data elements associated with each screen or GUI, or reference the data dictionary.*

## 9. Attachment A – Approval Signatures

This section is used to document the approval of the System Design Document. The review should be conducted face to face where signatures can be obtained ‘live’ during the review. If unable to conduct a face-to-face meeting then it should be held via LiveMeeting and concurrence captured during the meeting. The Scribe should add /es/name by each position cited. Example provided below.

The Business Sponsor and Project Manager are required to sign.

---

Signed:

Date:



Director, Veterans Transportation Program, CBO  
Business Sponsor

---

Signed:

Date:



Project Manager

## A. Additional Information

*Attach any addition information that supplements the design specification.*

### A.1. Identification of Technology and Standards

*Identify the system and software which apply to the SDD, including: identification number(s), title(s), abbreviation(s), version number(s), and release number(s). Identify all standards (e.g., American National Standards Institute [ANSI], International Organization for Standardization [ISO], Institute of Electrical and Electronics Engineers [IEEE], etc.).*

### A.2. Constraining Policies, Directives and Procedures

*Identify any constraints or requirements placed on this document by policies, directives, or procedures.*

### A.3. Requirements Traceability Matrix

*Include an RTM that traces modules and data structures to the software requirements. A reference to the location of the RTM is also acceptable.*

### A.4. Software Component Traceability

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
EPIC02 (BN18/19/20)		Beneficiary Travel Claim Interface BTSSS Administration Interface
OWNR18.02	S02.01	Beneficiary Travel Claim Interface BTSSS Administration Interface
OWNR18.03	S02.02	Beneficiary Travel Claim Interface BTSSS Administration Interface
OWNR19.02	S02.03	Beneficiary Travel Claim Interface BTSSS Administration Interface
OWNR19.03	S02.04	Beneficiary Travel Claim Interface BTSSS Administration Interface
OWNR20.02	S02.05	Beneficiary Travel Claim Interface BTSSS Administration Interface
EPIC03 (BN n/a)		Beneficiary Travel Claim Interface Eligibility Verification
N/A	S03.01	Beneficiary Travel Claim Interface

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
N/A	S03.02	Beneficiary Travel Claim Interface
N/A	S03.03	Eligibility Verification
N/A	S03.04	Eligibility Verification
EPIC04 (BN n/a)		Beneficiary Travel Claim Interface BTSSS Administration Interface
N/A	S04.01	BTSSS Administration Interface
N/A	S04.02	Beneficiary Travel Claim Interface BTSSS Administration Interface
N/A	S04.03	Beneficiary Travel Claim Interface BTSSS Administration Interface
EPIC05 (BN2/3)		Beneficiary Travel Claim Interface Profile Manager Eligibility Verification
OWNR2.02	S05.01	Profile Manager
OWNR2.02	S05.02	Profile Manager
OWNR2.01	S05.03	Beneficiary Travel Claim Interface
OWNR2.02	S05.04	Eligibility Verification
OWNR2.02	S05.05	Eligibility Verification
OWNR2.02	S05.06	Profile Manager
OWNR2.01	S05.07	Beneficiary Travel Claim Interface
OWNR2.01	S05.08	Profile Manager
N/A	S05.09	Profile Manager
OWNR2.04	S05.10	Profile Manager
OWNR2.04	S05.11	Profile Manager



Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
N/A	S05.12	Profile Manager
N/A	S05.13	Profile Manager
N/A	S05.14	Profile Manager
OWNR2.06	S05.15	Beneficiary Travel Claim Interface
OWNR3.01/3.02/3.03	S05.16	Profile Manager
N/A	S05.17	Profile Manager
N/A	S05.18	Profile Manager
N/A	S05.19	Eligibility Verification
N/A	S05.20	Profile Manager
EPIC06 (BN6)		Beneficiary Travel Claim Interface Profile Manager
OWNR6.04	S06.01	Beneficiary Travel Claim Interface
OWNR6.04	S06.02	Profile Manager
OWNR6.04	S06.03	Profile Manager
N/A	S06.04	Profile Manager
EPIC07 (BN7)		Eligibility Verification Beneficiary Travel Claim Interface Profile Manager Reimbursement Calculator Encounter Information Retriever
OWNR7.01	S07.01	Eligibility Verification
OWNR7.02	S07.02	Eligibility Verification
N/A	S07.03	Eligibility Verification
N/A	S07.04	Eligibility Verification

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
N/A	S07.05	Beneficiary Travel Claim Interface
N/A	S07.06	Encounter Information Retriever
N/A	S07.07	Reimbursement Calculator
N/A	S07.08	Eligibility Verification
N/A	S07.09	Beneficiary Travel Claim Interface
EPIC08 (BN n/a)		Profile Manager Eligibility Verification Encounter Information Retriever Reimbursement Calculator Business Rules Management
N/A	S08.01	Encounter Information Retriever
N/A	S08.02	Reimbursement Calculator
N/A	S08.03	Eligibility Verification
N/A	S08.04	Eligibility Verification
N/A	S08.05	Eligibility Verification
N/A	S08.06	Eligibility Verification
N/A	S08.07	Profile Manager
N/A	S08.08	Profile Manager
N/A	S08.09	Profile Manager
N/A	S08.10	Profile Manager
N/A	S08.11	Business Rules Management
N/A	S08.12	Business Rules Management
N/A	S08.13	Business Rules Management
N/A	S08.14	Business Rules Management

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
EPIC09 (BN4/5/8)		Beneficiary Travel Claim Interface Profile Manager Eligibility Verification Encounter Information Retriever
N/A	S09.01	Beneficiary Travel Claim Interface
OWNR4.01	S09.02	Beneficiary Travel Claim Interface
OWNR4.02	S09.03	Profile Manager
OWNR4.03	S09.04	Profile Manager
OWNR4.05	S09.05	Beneficiary Travel Claim Interface
OWNR4.07	S09.06	Encounter Information Retriever
N/A	S09.07	Beneficiary Travel Claim Interface
N/A	S09.08	Beneficiary Travel Claim Interface
OWNR4.08	S09.09	Profile Manager
N/A	S09.10	Beneficiary Travel Claim Interface
N/A	S09.11	Beneficiary Travel Claim Interface
OWNR4.09	S09.12	Beneficiary Travel Claim Interface
OWNR4.12	S09.13	Beneficiary Travel Claim Interface
OWNR4.13	S09.14	Beneficiary Travel Claim Interface
OWNR4.14	S09.15	Beneficiary Travel Claim Interface
OWNR4.15	S09.16	Beneficiary Travel Claim Interface
N/A	S09.17	Beneficiary Travel Claim Interface
N/A	S09.18	Beneficiary Travel Claim Interface
OWNR5.01	S09.19	Beneficiary Travel Claim Interface

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
OWNR5.02	S09.20	Beneficiary Travel Claim Interface
OWNR5.03	S09.21	Beneficiary Travel Claim Interface
OWNR5.04	S09.22	Beneficiary Travel Claim Interface
OWNR5.05	S09.23	Beneficiary Travel Claim Interface
OWNR5.06	S09.24	Beneficiary Travel Claim Interface
OWNR5.07	S09.25	Profile Manager
N/A	S09.26	Profile Manager
N/A	S09.27	Profile Manager
N/A	S09.28	Profile Manager
N/A	S09.29	Profile Manager
OWNR8.01	S09.30	Eligibility Verification
OWNR8.02	S09.31	Encounter Information Retriever
EPIC10 (BN6)		Beneficiary Travel Claim Interface Profile Manager
OWNR6.01	S10.01	Profile Manager
OWNR6.01	S10.02	Profile Manager
OWNR6.01	S10.03	Profile Manager
OWNR6.01	S10.04	Beneficiary Travel Claim Interface
OWNR6.01	S10.05	Profile Manager
N/A	S10.06	Beneficiary Travel Claim Interface
OWNR6.03	S10.07	Profile Manager
OWNR6.03	S10.08	Profile Manager

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
OWNR6.03	S10.09	Beneficiary Travel Claim Interface
OWNR6.03	S10.10	Profile Manager
OWNR6.03	S10.11	Profile Manager
OWNR6.03	S10.12	Profile Manager
OWNR6.03	S10.13	Profile Manager
OWNR6.05	S10.14	Profile Manager
OWNR6.06	S10.15	Profile Manager
EPIC11 (BN17)		Business Rules Management
OWNR17.01	S11.01	Business Rules Management
OWNR17.01	S11.02	Business Rules Management
OWNR17.01	S11.03	Business Rules Management
EPIC12 (BN16)		Workflow Management and Reporting
OWNR16.01/16.02	S12.01	Workflow Management and Reporting Profile Manager
OWNR16.03	S12.02	Workflow Management and Reporting
OWNR16.04	S12.03	Profile Manager
N/A	S12.04	Workflow Management and Reporting
N/A	S12.05	Workflow Management and Reporting
EPIC13 (BN9)		Reimbursement Calculator Eligibility Verification Encounter Information Retriever Profile Manager
N/A	S13.01	Encounter Information Retriever
N/A	S13.02	Encounter Information Retriever

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
N/A	S13.03	Encounter Information Retriever
N/A	S13.04	Encounter Information Retriever
OWNR9.01	S13.05	Reimbursement Calculator
N/A	S13.06	Reimbursement Calculator
OWNR9.02	S13.07	Profile Manager
OWNR9.05	S13.08	Reimbursement Calculator
OWNR9.06	S13.09	Reimbursement Calculator
OWNR9.07	S13.10	Reimbursement Calculator
EPIC14 (BN10)		Reimbursement Calculator
OWNR10.01	S14.01	Reimbursement Calculator
OWNR10.02	S14.02	Reimbursement Calculator
N/A	S14.03	Reimbursement Calculator
EPIC15 (BN11/12)		Payment File Generator Encounter Information Retriever Beneficiary Travel Claim Interface
OWNR11.01	S15.01	Payment File Generator
OWNR11.02	S15.02	Encounter Information Retriever
OWNR11.03	S15.03	Beneficiary Travel Claim Interface
OWNR11.04	S15.04	Beneficiary Travel Claim Interface
OWNR12.01/12.02	S15.05	Payment File Generator
EPIC16 (BN14)		Profile Manager Reimbursement Calculator Beneficiary Travel Claim Interface
OWNR14.02	S16.01	Reimbursement Calculator

Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
OWNR14.03	S16.02	Reimbursement Calculator
OWNR14.03	S16.03	Reimbursement Calculator
N/A	S16.04	Profile Management
OWNR14.03	S16.05	Beneficiary Travel Claim Interface
EPIC17 (BN13)		Beneficiary Travel Claim Interface Appeal Submission Eligibility Verification
OWNR13.01	S17.01	Beneficiary Travel Claim Interface
OWNR13.02	S17.02	Beneficiary Travel Claim Interface
OWNR13.03	S17.03	Beneficiary Travel Claim Interface
OWNR13.04	S17.04	Beneficiary Travel Claim Interface
N/A	S17.05	Appeal Submission
EPIC18 (BN15)		Workflow Management and Reporting
OWNR15.01/15.02/15.04	S18.01	Workflow Management and Reporting
N/A	S18.02	Workflow Management and Reporting
N/A	S18.03	Workflow Management and Reporting
N/A	S18.04	Workflow Management and Reporting
N/A	S18.05	Workflow Management and Reporting
N/A	S18.06	Workflow Management and Reporting
N/A	S18.07	Workflow Management and Reporting
N/A	S18.08	Workflow Management and Reporting
N/A	S18.09	Workflow Management and Reporting



Bus Req (BN or Epic) ID (Unique Identifier)	Req ID (Unique Identifier)	Generic SW Component
N/A	S18.10	Workflow Management and Reporting
OWNR15.03	S18.11	Workflow Management and Reporting
N/A	S18.12	Workflow Management and Reporting
N/A	S18.13	Workflow Management and Reporting
N/A	S18.14	Workflow Management and Reporting
N/A	S18.15	Workflow Management and Reporting
N/A	S18.16	Workflow Management and Reporting
N/A	S18.17	Workflow Management and Reporting
N/A	S18.18	Workflow Management and Reporting
N/A	S18.19	Workflow Management and Reporting
N/A	S18.20	Workflow Management and Reporting

## **A.5. Packaging and Installation**

*Outline any special considerations for software packaging and installation.*

## **A.6. Design Metrics**

*Describe all metrics to be used during the design activity.*