**Collaborative Terminology Tooling & Data Management (CTT&DM) Server & Tooling**

**System Design Document**



**Department of Veterans Affairs**

**February 2016**

**Version 2.1**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 02/10/2016 | 2.1 | Update TRM List- Tables 23 & 24 | Engility Team |
| 09/15/2015 | 2.0 | Update latest version/combined Server and Tooling | Engility/Rosalind Seale/William McDonough |
| 03/18/2015 | 1.0 | Update version | Engility/Soni  Mirchandani |
| 02/20/2015 | 1.0 | Final draft | Engility Team |
| 01/05/2015 | 1.0 | Initial draft | Engility Team |

**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 the product is built, to reflect the as-built product.

**When to Complete Each Section of the SOD**

|  |  |  |
| --- | --- | --- |
| **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 resomces that occm at project initiation. This gives the enterprise an opportunity to evaluate IT investments before project teams are stood up and ftmding is allocated. Sections 1- 4 which discuss the high level design, and 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.

|  |  |  |
| --- | --- | --- |
| **Activity** | **New Capability (1)** | **Feature Enhancement** (2) |
| **Field Deployment (A)** | Yes | No |
| **Cloud/Web**  **Deployment (B)** | Yes | No |
| **Mobile Application (C)** | Yes | No |

**Table of Contents**

**1. Introduction ......................................................................................... 1**

**1.1. Scope ................................................................................................................ 2**

**1.2. User Profiles (Tooling) .................................................................................... 3**

**2. Background ......................................................................................... 4**

**2.1. Overview of the System .................................................................................. 4**

**2.2. Overview of the Business Process ................................................................ 4**

**2.3. Overview of the Significant Requirements .................................................... 6**

**2.3.1. Overview of Functional Workload/Performance Requirements .......... 8**

**2.3.2. Overview of Operational Requirements ................................................. 8**

**2.3.3. Overview of the Technical Requirements .............................................. 8**

**2.3.4. Overview of the Security or Privacy Requirements .............................. 9**

**2.3.5. Overview of System Criticality and High Availability Requirements... 9**

**2.3.6. Single Sign-on Requirement ................................................................. 10**

**2.3.7. Requirement for Use of Enterprise Portals (Tooling) ......................... 10**

**2.3.8. Special Device Requirements ............................................................... 10**

**2.3.9. Legacy System Retirement ................................................................... 10**

**3. Conceptual Design............................................................................ 11**

**3.1. Conceptual Application Design .................................................................... 11**

**3.1.1. Application Context ............................................................................... 14**

**3.1.2. High-Level Application Design ............................................................. 15**

**3.1.3. Application Locations ........................................................................... 18**

**3.2. Conceptual Data Design................................................................................ 18**

**3.2.1. Project Conceptual Data Model ............................................................ 18**

**3.2.2. Database Information ............................................................................ 19**

**3.2.3. User Interface Data Mapping ................................................................ 19**

3.2.3.1. Application Screen Interface ............................................................... 19

3.2.3.2. Application Report Interface ................................................................ 20

3.2.3.2.1. Insert Name of Report ................................................................................. 20

3.2.3.3. Unmapped Data Element .................................................................... 20

**3.3. Conceptual Infrastructure Design (Server).................................................. 20**

**3.3.1. Process and Activity Monitoring Server System Criticality and High**

**Availability ........................................................................................................... 21**

**3.3.2. Special Technology ............................................................................... 21**

**3.3.3. Technology Locations........................................................................... 22**

**3.3.4. Conceptual Infrastructure Diagram (Server) ....................................... 23**

3.3.4.1. Location of Environments and External Interfaces .............................. 23

3.3.4.2. Conceptual Production String Diagram ............................................... 23

**4. System Architecture ......................................................................... 24**

**4.1. Hardware Architecture (Server) .................................................................... 24**

**4.2. Software Architecture (Tooling) ................................................................... 27**

**4.3. Network Architecture (Server) ...................................................................... 29**

**4.4. Service Oriented Architecture/ESS (Tooling) .............................................. 30**

**4.5. Enterprise Architecture (Tooling)................................................................. 31**

**5. Data Design ....................................................................................... 34**

**5.1. Data Base Management System (DBMS) Files ............................................ 42**

**5.2. Data View ........................................................................................................ 42**

**6. Detailed Design ................................................................................. 43**

**6.1. Hardware Detailed Design............................................................................. 44**

**6.2. Software Detailed Design .............................................................................. 46**

**6.2.1. Conceptual Design ................................................................................ 46**

6.2.1.1. Product Perspective ............................................................................ 46

6.2.1.1.1. User Interfaces ............................................................................................. 46

6.2.1.1.2. Hardware Interfaces..................................................................................... 46

6.2.1.1.3. Software Interfaces ...................................................................................... 46

6.2.1.1.4. Communications Interfaces ......................................................................... 46

6.2.1.1.5. Memory Constraints .................................................................................... 47

6.2.1.1.6. Special Operations ....................................................................................... 47

6.2.1.2. Product Features ................................................................................ 47

6.2.1.3. User Characteristics............................................................................ 47

6.2.1.4. Dependencies and Constraints ........................................................... 47

**6.2.2. Specific Requirements .......................................................................... 48**

6.2.2.1. Database Repository........................................................................... 48

6.2.2.2. System Features ................................................................................. 49

6.2.2.3. Design Element Tables ....................................................................... 50

6.2.2.3.1. Routines (Entry Points) ............................................................................... 50

6.2.2.3.2. Templates..................................................................................................... 51

6.2.2.3.3. Bulletins....................................................................................................... 52

6.2.2.3.4. Data Entries Affected by the Design ........................................................... 53

6.2.2.3.5. Unique Record(s)......................................................................................... 53

6.2.2.3.6. File or Global Size Changes ........................................................................ 53

6.2.2.3.7. Mail Groups ................................................................................................. 54

6.2.2.3.8. Security Keys............................................................................................... 55

6.2.2.3.9. Options......................................................................................................... 56

6.2.2.3.10. Protocols .................................................................................................... 58

6.2.2.3.11. Remote Procedure Call (RPC)................................................................... 59

6.2.2.3.12. Constants Defined in Interface .................................................................. 59

6.2.2.3.13. Variables Defined in Interface................................................................... 60

6.2.2.3.14. Types Defined in Interface ........................................................................ 60

6.2.2.3.15. GUI ............................................................................................................ 60

6.2.2.3.16. GUI Classes ............................................................................................... 60

6.2.2.3.17. Current Form ............................................................................................. 60

6.2.2.3.18. Modified Form........................................................................................... 60

6.2.2.3.19. Components on Form ................................................................................ 61

6.2.2.3.20. Events ........................................................................................................ 61

6.2.2.3.21. Methods ..................................................................................................... 61

6.2.2.3.22. Special References..................................................................................... 61

6.2.2.3.23. Class Events............................................................................................... 61

6.2.2.3.24. Class Methods ........................................................................................... 61

6.2.2.3.25. Class Properties ......................................................................................... 61

6.2.2.3.26. Uses Clause................................................................................................ 62

6.2.2.3.27. Forms ......................................................................................................... 62

6.2.2.3.28. Functions ................................................................................................... 62

6.2.2.3.29. Dialog ........................................................................................................ 64

6.2.2.3.30. Help Frame ................................................................................................ 65

6.2.2.3.31. HL7 Application Parameter....................................................................... 65

6.2.2.3.32. HL7 Logical Link ...................................................................................... 66

6.2.2.3.33. COTS Interface.......................................................................................... 67

**6.3. Network Detailed Design ............................................................................... 67**

**6.4. Security and Privacy...................................................................................... 68**

**6.4.1. Security................................................................................................... 68**

**6.4.2. Privacy .................................................................................................... 68**

**6.5. Service Oriented Architecture/ESS Detailed Design .................................. 69**

**6.5.1. Service Description for <Consumed Service Name>.......................... 70**

**6.5.2. Service Design for <Provided Service Name> .................................... 70**

6.5.2.1. Introduction ......................................................................................... 70

6.5.2.1.1. Purpose and Scope of Service ..................................................................... 70

6.5.2.1.2. Links to Other Documents........................................................................... 70

6.5.2.2. Service Details .................................................................................... 70

6.5.2.2.1. Service Identification................................................................................... 70

6.5.2.2.2. Service Versions .......................................................................................... 71

6.5.2.2.3. Summary of Design and Platform Details ................................................... 71

6.5.2.2.3.1. SOA Pattern(s) Implemented ................................................................ 71

6.5.2.2.3.2. COTS Platform Vendor Names and Versions for Hosting Platform .... 71

6.5.2.3. Dependencies ..................................................................................... 71

6.5.2.4. Service Design Details ........................................................................ 71

6.5.2.4.1. Interface Technical Specs ............................................................................ 72

6.5.2.4.1.1. Service Invocation Type ....................................................................... 72

6.5.2.4.1.2. Service Interface Type .......................................................................... 72

6.5.2.4.1.3. Service Name ........................................................................................ 72

6.5.2.4.1.4. Interface ................................................................................................ 72

6.5.2.4.1.5. End Points ............................................................................................. 72

6.5.2.4.1.6. Operations or Methods.......................................................................... 72

6.5.2.4.1.7. Message Schemas ................................................................................. 72

6.5.2.4.2. Information Model....................................................................................... 72

6.5.2.4.2.1. Class Diagram and Description of Entities Involved............................ 72

6.5.2.4.2.2. Mappings from ELDM to Standards Based Schemas........................... 72

6.5.2.4.3. Behavior Model (AKA Use Case Realization)............................................ 72

6.5.2.4.3.1. Use Cases (Use Case Model) ................................................................ 73

6.5.2.4.3.2. Interaction Diagrams............................................................................. 78

6.5.2.5. Gap Analysis ....................................................................................... 80

6.5.2.5.1. Variances from Enterprise Target Architecture........................................... 90

6.5.2.5.2. Variances from SLDs .................................................................................. 90

6.5.2.5.3. Variances from Standards and Policies ....................................................... 91

6.5.2.5.4. Justification for Exceptions and Mitigation................................................. 91

**7. External System Interface Design ................................................... 91**

**7.1. Interface Architecture .................................................................................... 91**

**7.2. Interface Detailed Design .............................................................................. 91**

**8. Human-Machine Interface ................................................................ 91**

**8.1. Interface Design Rules .................................................................................. 91**

**8.2. Inputs .............................................................................................................. 91**

**8.3. Outputs ........................................................................................................... 91**

**8.4. Navigation Hierarchy ..................................................................................... 91**

**8.4.1. Screen [x.1] ............................................................................................ 91**

**8.4.2. Screen [x.2] ............................................................................................ 91**

**8.4.3. Screen [x.3] ............................................................................................ 91**

**9. Attachment A – Approval Signatures.............................................. 92**

**A. Additional Information ...................................................................... 93**

**A.1. Identification of Technology and Standards ............................................... 93**

**A.2. Constraining Policies, Directives and Procedures ..................................... 93**

**A.3. Requirements Traceability Matrix (RTM) ..................................................... 93**

**A.4. Packaging and Installation............................................................................ 93**

**A.5. Design Metrics ............................................................................................... 93**

**1. Introduction**

Collaborative Terminology Tooling & Data Management (CTT&DM) is a consolidation of different suites of applications that will be centralized under one system. This consolidated solution will provide a robust Technical Terminology Service (TTS) to standardize terminologies across the various medical coding systems. All systems and software will be open source. The support for these terminologies and support systems will be accessed through the Collaborative Terminology Servers. These will include the following systems and enhancements:

x Standard Code System (SCS) Terminology standards produced by a Standards

Development Organization (SDO)

x Veteran Health Information Systems and Technology Architecture(VistA) Lexicon standard reference for clinical terminology across the Department of Veterans Affairs (VA) containing coding systems: Clinical Procedural Terminology (CPT) & CPT Modifier

x Healthcare Common Procedure Coding System (HCPCS)

x International Classification of Diseases-9 (ICD-9)

x Systematized Nomenclature of Medicine –Clinical Terminology (SNOMED CT)

x Veteran Heath Administration Terminology (VHAT), A Standard Code System created and maintained by Standard & Terminology Services (STS), specifically for use in enterprise VA systems. VHAT provides Allergies/Adverse Reactions, Vitals, Orders (partial), Demographics (partial), Pharmacy, and TIU documents titles.

x Map Sets. Associations between concepts from any two standard code systems or within the same standard code system

x General Equivalence Mappings (GEMs). A set of map sets used to translate between the ICD-9-CM and ICD-10-CM/PCS code systems. They are created by the National Center for Health Statistics (NCHS) and the Center or Medicare & Medicaid Studies (CMS).

x Logical Observation Identifier Names and Codes (LOINC). A database and universal standard for identifying medical laboratory observations.

x VA Enterprise Technology Services is a Java/Oracle based development which includes: Terminology Deployment Services (TDS) that allows terminologist to prepare, test, deploy, and troubleshoot terminology to VA systems.

x New Term Rapid Turnaround (NTRT) Web Applications allow field-users to request new terms, and track submissions through a web interface.

x Terminology Services allow other applications to access terminology content.

x VA Unique Identifiers (VUID) Services provide VUIDs through an Application

Programming Interface (API) or a web interface.

x SOLOR, an integration of SNOMED, LOINC, and RxNorm

x SNOMED VA extension

**1.1. Scope**

The scope of this document includes the functionalities and enhancements to replace the existing VA Enterprise Terminology Service (VETS) system to include web-based access along with the integration of the Intemational Health Tenninology Standard Development Organization (IHTSDO) Workbench, and Infonnatics Architecture Acceleration (ISAAC) toolset. This new web-based tool aims to standardize terminology content used across the Veterans Health Administration (VHA), and will be accessible by clinicians, and tenninologists responsible for updating clinical tenninology in VistA, ultimately giving them the ability to update tenninology in real time across disparate systems.

**Table 1 Overarching CTT&DM ReQUirements**

|  |  |  |
| --- | --- | --- |
| # | **Functionality/Services** | **Description** |
| 1 | Terminology Server (Run time) | Provide programmatic access to terminology components. Host terminology from the following systems:  SNOMED CT  SNOMED US Extensions SNOMED CT Spanish Edition SNOMED VA Extension LOINC  RxNORM  ICD-9-CM ICD-10-CM CPT/HCPCS DRG  VHAT  SOLOR |
| 2 | Continuous Integration Server  (Content Management Service) | Provide the ability to manage the developmental workflow by providing an automated process from check-ouUcheck-in of code to release.  Prototypical examples of such continuous integration services include: Apache Continuum, Bamboo, Hudson, Jenkins, and Team City. |
| 3 | ArtifacUProject Repository | Release repository for project dependencies. All project artifacts shall be managed by an Apache Maven compatible repository manager and extend Apache Maven with pluqin extensions to manage terminology project processing, quality determination, release, and deployment.  Artifactory Apache Archiva and Sonotype Nexus are prototypical examples of artifact repository services.  Apache Ant is another software build tool that compares to Maven and may be used. |

|  |  |  |
| --- | --- | --- |
| # | **Functionality/Services** | **Description** |
| 4 | Version/Source Control | Provide a distributed version control system, where developers will have the ability to work on the same code/project at the same time.  Provide version control for all ETS content. Each file will be tracked, any modifications made, additions, deletions, or movement of files will receive a version number upon commit. |
| 5 | Workflow Management & Rules  Engine | Support the business life cycle from authoring through execution to monitoring and management - utilizing a workflow management & rules engine.  Workflow Management Service deployed on a Java EE ?/Java 8 SE server. Workflow Service will support both Representational State Transfer (REST) Application Program Interface (API) and Web Services Human Task (WS-Human Task) Management. |
| 6 | Component Request Service | Provide the ability to allow users to enter modification requests for terminology that can be tracked and managed globally. |
| 7 | Quality Management | Provide continuous inspection of code quality utilizing business rules from local validation through release validation. |
| 8 | Database Server | The Database Server will be a Oracle SQL Server |
| 9 | Issue Tracking Service | Provide the ability to enter and track defects |
| 10 | Service Desk | Provide the ability for users to request assistance from Information Technology (IT) via a ticketing system |

**1.2. User Profiles (Tooling)**

The following are the two groups of users for the tooling application.

• Primaty Users: Standards and Tetminology Services (STS) and Knowledge Based

Systems (KBS) Staff

• Secondaty Users: Mobile Health and Others (Clinical Inf01mation System/Anesthesia

Record Keeper/Nmsing, etc.)

To the extent that new functionality or changes that affect the usability of the tooling application are introduced, STS, KBS, Mobile Health, and Other Staff Users will require training to ensme that each understands the steps needed to successfully execute their responsibilities.

**2. Background**

**2.1. Overview of the System**

The Collaborative Tetminology Development Tooling environment will initially allow tetminology mapping, browse/search tetminology, workflow automation, and tetminology editing.

The application server will be a platf01m for developing and deploying multi-tier distributed enterprise applications. The server centralizes application services, such as Web server functionality, business components, and access to backend enterprise systems. It should utilize technologies, such as caching and connection pooling, to improve resource use and application perf01mance, as well as other components/tools for improved perf01mance.

The Collaborative Tetminology Development Tooling is being developed because the VA needs an approved resource tetminology plan that will be successful and meet the goals of interoperability, standardization, and computability as mandated by Congress in the National Defense Authorization Act (NDAA).

The environment will improve workflow, data collection, and quality of operations.

**2.2. Overview of the Business Process**

The inf01mation shown in Table 2 provides an overview of the major categories of business process, and Figure 1 is a simplified business process flow diagram. Rather than providing a full detailed enumeration of requirements, the intention of the SDD is to show the major business process. This sets the architecture direction and drives the design depicted in the following sections. For additional details regarding the functional requirements, please refer to the Business Requirements Document (BRD).

**Table 2· Business Process**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Administrative | The ability to manage terminology source content in the terminology environment. |
| Terminology Search | The ability to effectively search and filter for terminology content based search criteria. |
| Terminology Workflow Process | The ability to support workflow and business processes for developing, reviewing, and terminology update requests. |
| Terminology Updates | The ability to create, view, update, and retire terminology. |

Notify UserOf

Assigned

Request

Editor is

Notified of

Rejected Status

User is notified of Approval

Request

Reviewer

User Review

Request

Request Submitted to Reviewer

Requestis marked as "Assigned"

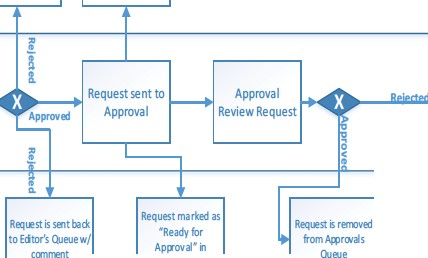
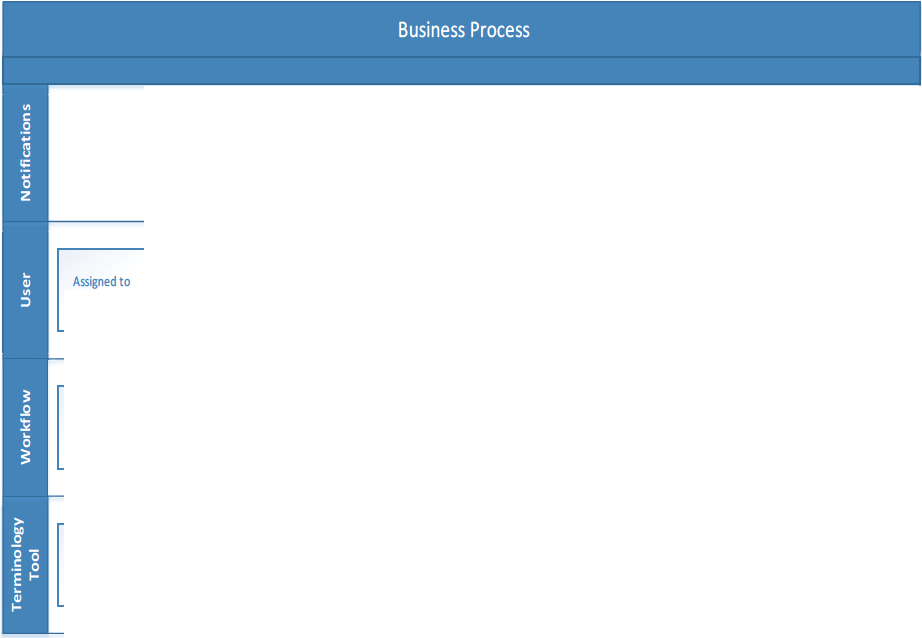
as "Reviewed"

**Awovers Queue**

RequestS<nt bade

Ed tor's

**canrrent**



Editor Change Request Initiated

Request Publislled Terminology ToolUpdated

**Figure 1: Tooling Business Processes Diagram**

CTIDM Functional Model v1.0



t

8a.

:J

*(/)*

c

.Q

*E*

":J'

0

:J

.E

c

0

()



**2.3.**

Send Change<

No

ASsigned Log Issue

Issues

Figure 2: Server Business Processes Diagram

**Overview of the Significant Requirements**

The inf01mation shown in the table below provides an overview of the major categories of functional requirements. Rather than providing a fully detailed enumeration of requirements, the intention of the SDD is to show the major fi.mctionality; this sets the architecture direction and drives the design depicted in the following sections.

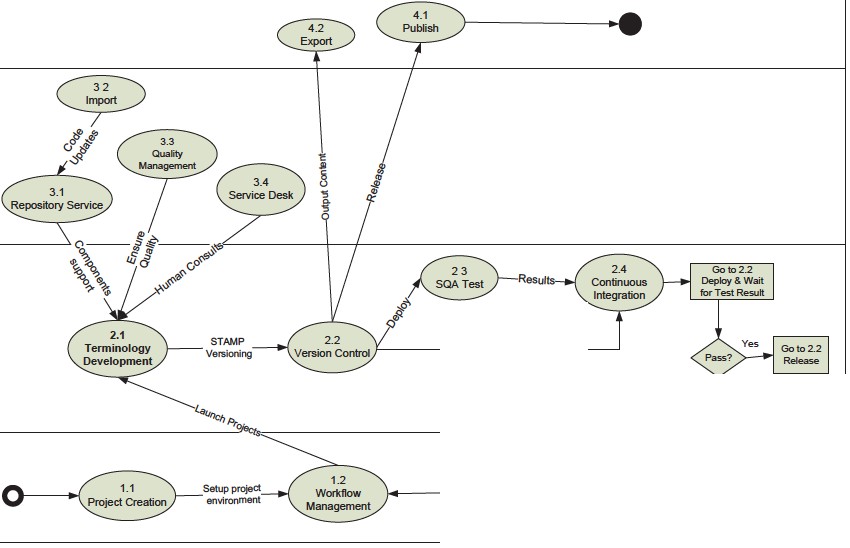


Table 3: Functional Requirements

|  |  |
| --- | --- |
| 10 | Requirement |
| BRD ETS-F-121 | Provide capability to send user alert notifications. |
| BRD ETS-F-22 | Provide Identity and Access Management  Service. |
| BRD ETS-F-123 | Provide capability for human consults. |
| BRD ETS-F-147 | Provide capability to accept, process, integrate, and version electronic releases of terminologies from SDO, the National Library Medicine (NLM) and other authoritative sources. |

|  |  |
| --- | --- |
| **ID** | **Requirement** |
| BRD ETS-F-151 | Provide capability to export terminology content and knowledge artifacts in a variety of formats for internal and external consumption. Exports must support versioning error checking, tracking, and published/scribe paradigms. |
| BRD ETS-F-149 | Provide capability that will support searchable, shareable Extensible Mark-up Language  (XML) structures. |
| BRD ETS-F-178 | Provide capability to support a mechanism for integrating data from the terminology tooling. |
| BRD ETS-F-163 | Provide capability to support mapping between and among terms and knowledge artifacts. |
| BRD ETS-F-9 | Provide capability to a workflow management service able to create, load, and execute rules. Also to coordinate workflows between Enterprise Terminology Service ETS client applications. |
| OWNR1993 | Provide capability with a taxonomy view that must support displaying concepts parentage, when more than one parent is present and enabling navigation of the entire parentage, and lineage, of a concept. |
| BRD ETS-F-66 | Provide a unique 10 for Staus,Time Author Module Path (STAMP) versioning for all enterprise terminology services, content change; map sets, rules, refsets, concepts, and definitions. |
| BRD ETS-199 | Provide capability to support description logic in the Web Ontology Language (OWL 2 EL) profile. |
| 1 | Provide capability that shall allow administrators to set up and modify user information and permissions. |
| 2 | Provide capability to allow users the ability to add, retire, and edit new terms, relationships, and attributes to represented terminologies. |
| 3 | Provide capability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. |

**2.3.1. Overview of Functional Workload/Performance Requirements**

The following table is an overview of the functional workload/perf01mance requirements.

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

|  |  |
| --- | --- |
| **ID** | **Requirement** |
| 1 | The web application components of the system are expected to support 5,000 registered users, and 600 concurrent users. This includes STS staff, KBS staff, contractors, field staff, etc. |
| 2 | Editing environments that support classification must be able to classify the entire terminology in 30 seconds or less, and to incrementally classify additive changes to the terminology in 500 milliseconds or less. |
| 3 | Reset environments must support Reset computation of a 150,000 member set, and computation of parent Resets in less than 10 seconds. |

**2.3.2. Overview of Operational Requirements**

The table below lists the operational requirements for the CTT&DM project.

**T**a**bl**e **5 0merat1onaI ReQUirements**

|  |  |
| --- | --- |
| **ID** | **Requirement** |
| 1 | The system shall remain responsive at all times, with no more than a 2 second lag between user action, and system response to action. |
| 2 | System response times and page load times shall be consistent with VA  standards. |
| 3 | Maintenance, including maintenance of externally developed software incorporated into the application, shall be scheduled during off peak hours or in conjunction with relevant maintenance schedules. |
| 4 | 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 within 30 minutes of the occurrence. 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. |
| 5 | Provide a real-time monitoring solution during the maintenance windows or when technical issues/problems occur which may require a preventative back up. |
| 6 | Notification of scheduled maintenance periods that require the service to be offline or that may degrade system performance shall be disseminated to the user community a minimum of 48 hours prior to the scheduled event. |
| 7 | The system shall be 99.9% available Monday-Friday, 6:00AM ET to 3:00AM ET. The system shall be 98% available on an annual basis. |

**2.3.3. Overview of the Technical Requirements**

The table below provides a brief overview of the CTT&DM technical requirements.

**Tabl**e **6 T**ec**hm**.**ea**I **ReqUirements**

|  |  |
| --- | --- |
| **ID** | **Requirement** |
| 1 | Section 508-compliance:  All end user interfaces must be in compliant with VA handbook 6102, Section  508 and directives for website design usability. |
| 2 | Integration with other VA systems:  CTI&DM ability to integrate with other internal VA systems is one of the key drivers to the overall technical architecture. The solution must have the capacity to utilize the current infrastructure and enterprise standards and services to communicate and interface with other internal systems. |
| 3 | International Electrochemical Commission (IEC) 62304 Compliance:  CTI&DM will be in compliance with IEC 62304 for the development of medical device software. |

**2.3.4. Overview of the Security or Privacy Requirements**

The table below provides a brief overview of the CTT&DM security and privacy requirements.

**T**a**bl**e **7 Secunty ReqUirements**

|  |  |
| --- | --- |
| **ID** | **Requirement** |
| 1 | All VA security requirements will be adhered to. Based on Federal Information Processing Standard (FIPS) 199 and National Institute of Standards and Technology (NIST) SP 800-60, recommended Security Categorization is High. |
| 2 | 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 D (BRD). |
| 3 | 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. |

**2.3.5. Overview of System Criticality and High Availability**

**Requirements**

The sensitivity rating for the Collaborative Tooling system is specified in the following table.

**Table 8: Sensitivity Rating**

|  |  |  |
| --- | --- | --- |
| **Category** | **Sensitivity Definition** | **Rating** |
| Confidentiality | Protection from unauthorized disclosure of information. | Moderate |
| Integrity | Protection from unauthorized, unanticipated, or unintentional modification or destruction of information. | Moderate |

|  |  |  |
| --- | --- | --- |
| **Category** | **Sensitivity Definition** | **Rating** |
| Availability | Ability of the system to ensure that system resources (data, computer programs and equipment, network connectivity) are accessible and operational, at the required level of performance, when they are needed. | High |

The Collaborative Tooling system will be hosted in the following location(s): Location: Primaty - Austin Inf01mation Technology Center (AITC).

The Collaborative Tooling system is expected to be available at minimum 95% of the time with a

target goal of 99%.

The Collaborative Tooling system is mission critical system and full disaster recovety setup in a secondaty location is needed.

**2.3.6. Single Sign-on Requirement**

For the Collaborative Tooling system, The Project Inf01mation System and Management Environment (PRISME) shall provide Collaborative Tooling users with a single-sign-on capability to ensure that users only log onto PRISME once. All role-based access control is managed through the single-sign-on capability.

**2.3.7. Requirement for Use of Enterprise Portals (Tooling)**

The Collaborative Tooling system shall be accessible to as many as 600 concunent users, including public use. The Collaborative Tooling system shall consist of both a stand-alone application and entetprise portal based Web applications. The standalone systems pull their data from the centralized system and resynchronize upon reconnection to that centralized system.

**2.3.8. Special Device Requirements**

Not applicable

**2.3.9. Legacy System Retirement**

The Collaborative Tooling system will lead to long-te1m retirement and integration of VETS and the other tetminology systems cunently in use. The table below lists the proposed Legacy retirements.

|  |  |  |
| --- | --- | --- |
| **Tab** | **le 9 ProposedLegacy Ret1**.**r** | **ements** |
| **Legacy System or**  **Legacy System Component** | **System Retired or**  **Workload Reduced** | **Quantify the Workload**  **Reduction** |
| VETS & other Terminology  Servers | Long term retirement and integration | Not available |

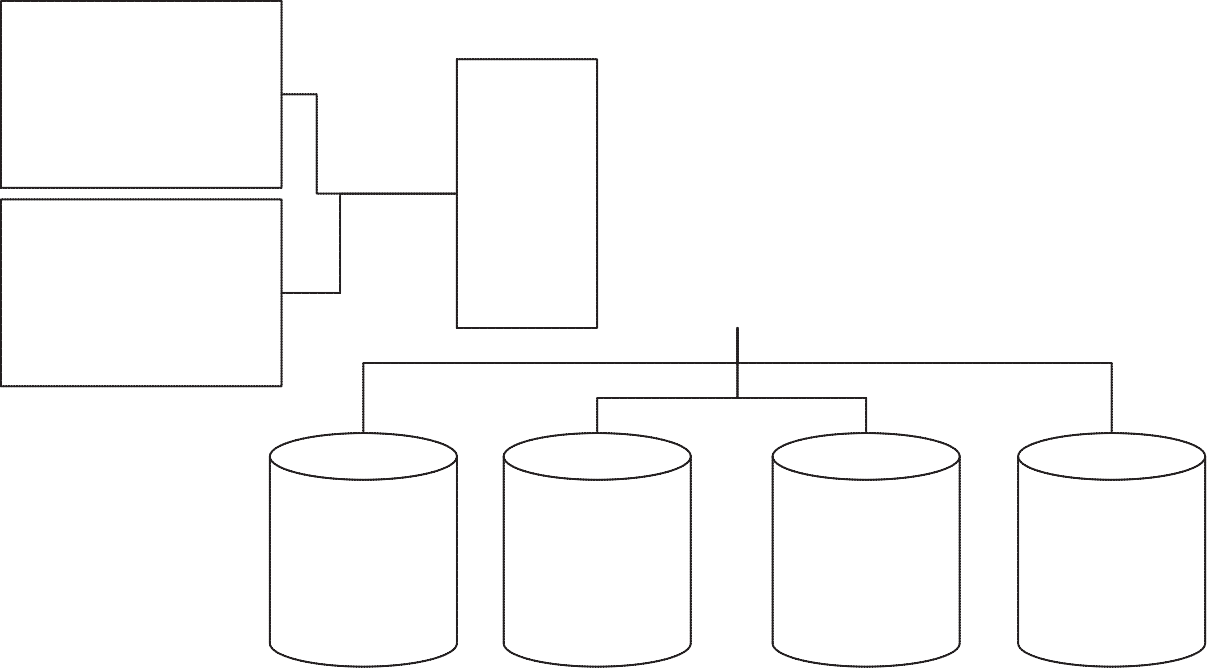
**3. Conceptual Design**

Section 3 provides a high-level overview of the Enterprise Terminology Services (ETS) (identified within the New Service Request [NSR] #20100904 Collaborative Terminology Developing Tooling) application design. The conceptual perspective provides a logical view of the approach; this is intended to show the overall approach and architecturally significant components.

**3.1. Conceptual Application Design**

The CTT&DM application design has been constructed to provide the required functionality as specified in the Requirements Specification Document (RSD) while maintaining compliance with the One-VA Technical Reference Model (TRM) and industry best practices.

The current system supports 8-10 users. The goal is to support 5,000 registered users and 600 concurrent users. Annual growth may be as high as 20% initially. The current system is in the pilot phase and consists of one application server and database server, with a separate webserver that resides on the physical application server. This can be expanded in an n-Tier model or virtualized to support a larger application audience once the pilot phase is complete. The current “As-Is” system is depicted in the figure below.



Termino logy

Brows er

Termino logy

Editor (TED)

(Brows er Interfa ce)

HTTPS

Web server

Termino logy Deployment

Services (TD S)

Web logic Application Server

Computerized Patient Record System (CPRS)

Unified

Medical

Langua ge System (UMLS)

Metathesa urus

Systemized

Nomenclature of

Medicine (SNOMED) Clinical Terms (CT)

Logical

Observa tion

Identi fiers, Names, And Codes (LOINC)

Ora cle Da ta base (Schemas and Da ta Imports)

**Figure 3: Current “As-Is” System**

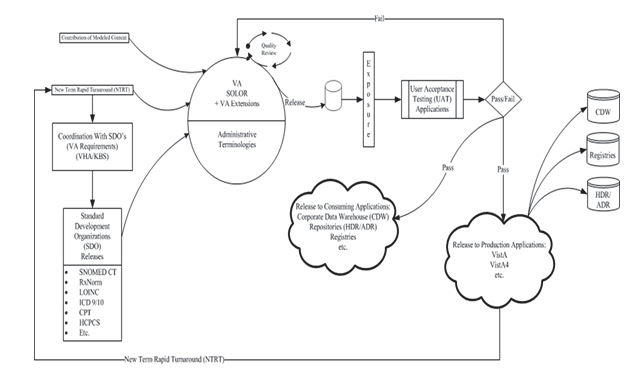
The CTT&DM conceptual design below is the “To-Be” view of the future application. Specifically, this architectural approach allows CTT&DM to meet the following requirements:

x Check for duplication daily through a QA process;

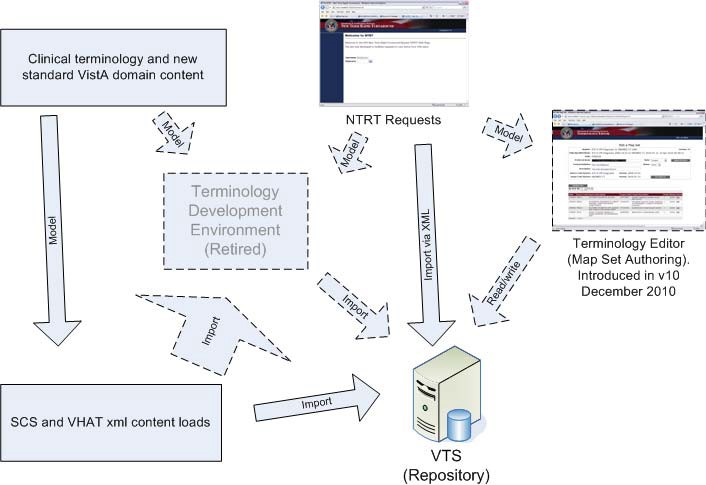
x Convert terms to any languages based on configuration;

x Allow users to be assigned multiple roles; and

x Allow the workflow status to be edited.



**Figure 4: Conceptual “To-Be” Application Design (1 of 2)**



**Figure 5: Conceptual “To-Be” Application Design (2 of 2)**

**3.1.1. Application Context**

The following figure represents the VETS database. VETS is one of the systems included within the CTT&DM project.

VHAT

STS Terrninology Browser

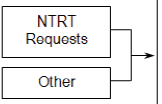
Web Services

Map Sets

scs

Manual Transform into Xlvt..

files



TDS

llll>Ort

&

versioning

Terminology

Reoositorv

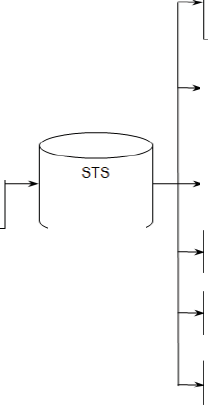
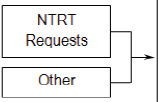
L...\_\_T\_os\_ H,

TED

NTRT VUID Server

stA

Figure 6: Application Context Diagram



The following tables outline the application context description.

Table 10 ObI"Ject

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Description | Interface  Name | Interface  System |
| N/A | N/A | N/A | N/A | N/A |

Table 11: Interfaces External to 01&T

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Name | Related Object | Input Messages | Output  Messages | External  Party |
| N/A | N/A | N/A | N/A | N/A | N/A |

Table 12: Interfaces Internal to OI&T

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Name | Related Object | Input Messages | Output  Messages | External  Party |
| VETS | VHA Enterprise Terminology Service | SCS,VHAT | SCS Data | Terminology Content and Knowledge Artifacts | VHA |

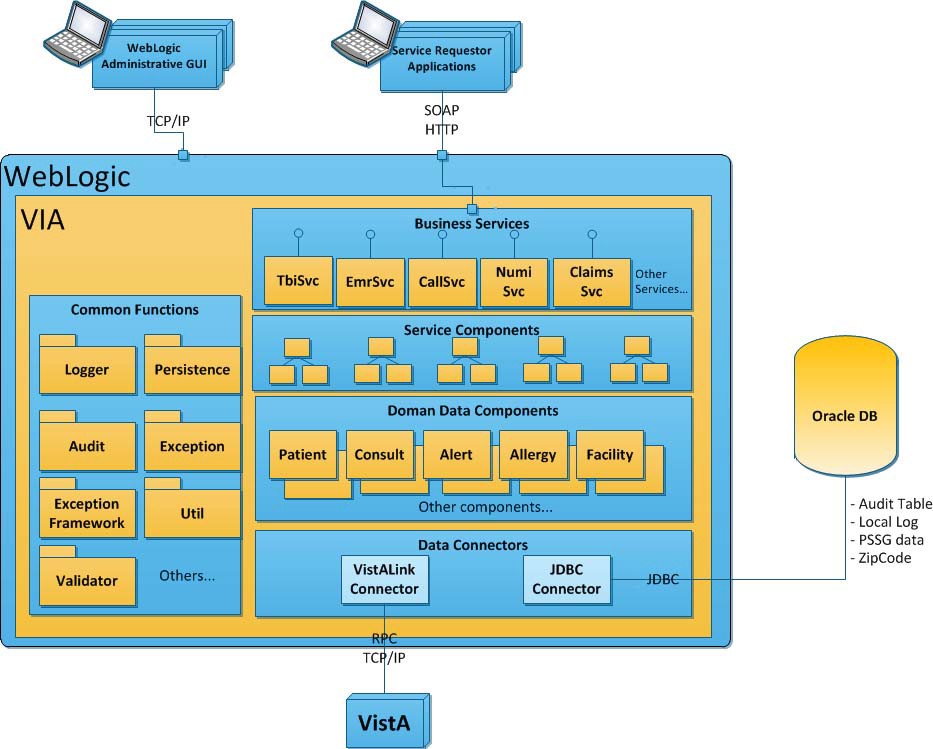
Table 13: Externally Shared Data Stores

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Data Stored | Owner | Access |
| N/A | N/A | N/A | N/A | N/A |

**3.1.2. High-Level Application Design**

PRISME provides a single-sign-on capability that allows users to authenticate once while maintaining all role-based access controls. The identified process flow diagram below allows users to send requests modeling assistance, provides configurable workflows capable of directing model review after creation, and a workflow inbox for users to manage their work.

A J2EE Platform running on a non-Windows dependent OS will be utilized.



**Figure 7: High-Level Application Design**

The following tables list the object/components in the high-level application design to be built or modified

**Table 14: Obiects *I* Comoonents to be Built or Modified**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Service or**  **Legacy Code** | **External Interface Name** | **External**  **Interface ID** | **Internal Interface Name** | **Internal**  **Interface ID** | **SOP Sections**  **1&2** |
| VETS | Veterans  Enterprise Terminology System | Suites of products that deliver standardized Terminology Content for use across the VA enterprise; including VistA and CHDR | Rapid and accurate issuance of  terms, updates of  existing terms, and deployment of solutions over a browser to multiple applications. | Veterans Enterprise Terminology System | VETS | Veterans Enterprise Terminolog y System | VETS | Being  Developed |
| VETS | Connected Health/Mobile Applications | Provides mobile access to VA healthcare information | Limited; applications are being developed | Connected Health/ Mobile Applications | Connected Health/ Mobile Applications | Connected Health/ Mobile Applications | Connected Health/ Mobile Applications | Being  Developed |
| cow | Corporate Data  Warehouse | National data repository that pulls from several VHA clinical and administrative systems, primary VistA. | Existing functionality | cow | cow | cow | cow | Being  Developed |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Name** | **Description** | **Service or**  **Legacy Code** | **External Interface Name** | **External**  **Interface ID** | **Internal Interface Name** | **Internal**  **Interface ID** | **SOP Sections**  **1&2** |
|  | Registries | Supports the maintenance of local and national registries for clinical and resource tracking of care for patients with certain  clinical conditions. | Existing functionality | Registries | Registries | Registries | Registries | Being  Developed |

**Table 15· Internal Data Stores**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Name** | **Data Stored** | **Steward** | **Access** |
| N/A | N/A | N/A | N/A | N/A |

**3.1.3. Application Locations**

The following tables describe the application locations and users for the CTT&DM applications.

**Table 16: Application Locations**

|  |  |  |  |
| --- | --- | --- | --- |
| **Application**  **Component** | **Description** | **Location at Which**  **Component is Run** | **Type** |
| VETS | Suite of products that deliver standardized terminology content for use across the VA enterprise | Austin Information Technology Center (AITC) | Client-facing servers that expose VA services |

**Table 17: Application Users**

|  |  |  |
| --- | --- | --- |
| **Application Component** | **Location** | **User** |
| VETS,TDS, STS | STS and KBS Staff | Primary User |
| VETS, TDS, STS | Mobile Health  (Connected Health)  Others (Clinical Information System/Anesthesia Record Keeper, Nursing, etc.) | Secondary User |

**3.2. Conceptual Data Design**

This section provides a high-level representation of the CTT&DM project's design approach for data.

**3.2.1. Project Conceptual Data Model**

Clinical and technical tooling to support a comprehensive Enterprise Terminology Service (ETS) will build on existing open-somce tenninology tooling projects cmTently either lmder development at, or in use at, the IHTSDO, the NLM, Kaiser Pe1manente, Sweden's National Release Center, Denmark's National Release Center, and the VHA.

Subjed Area 1 SUJbjed Area 2

Entity 1 Entity 2 Entity5 Entity 8

I I I I

Entity 6 Entity 9

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Entity 3 6rtity 4

E3 EntiJ 7 Ent\_J10

EJ B

Figure 8: Project Conceptual Data Model

3.2.2. Database Information

Table 18 Database nventory

|  |  |  |  |
| --- | --- | --- | --- |
| Database Name | Description | Type | Steward |
| STS Terminology  Repository | Oracle database housed at the Austin Information Technology Center | Interface | Terminology Deployment  Services |

3.2.3. User Interface Data Mapping

*NIA*

3.2.3.1. Application Screen Interface

*NIA*

Table 19 Screen Descnp·1r1on

|  |  |  |  |
| --- | --- | --- | --- |
| Graphical User  Interface (GUI) Field | Table (Database Table that field connects to) | Field (Field in Table that the GUI field connects to) | Comments |
| N/A | N/A | N/A | N/A |
| N/A | N/A | N/A | N/A |
| N/A | N/A | N/A | N/A |

3.2.3.2. Application Report Interface

*NIA*

3.2.3.2.1. Insert Name of Report

*NIA*

Table 20 Descn.p1f1on

|  |  |
| --- | --- |
| Report Column | Data Source |
| Patient | N/A |
| SSN | N/A |
| DoB | N/A |

3.2.3.3. Unmapped Data Element

*NIA*

**3.3. Conceptual Infrastructure Design (Server)**

The CTT&DM system will be composed of a variety of teclmology systems and those systems are widely available in the marketplace. The preference is to integrate existing systems rather than developing evetything from scratch. The teclmology systems that are available in the marketplace and that will be integrated as patis of the CTT&DM system include but are not limited to the following:

1. Tenninology server (open somce)

2. Continuous integration

3. Issue tracker

4. Version control

5. Business process management.

There are many options in the marketplace for the selection of the above teclmologies. The teclmologies may be open somce or proprietaty - with open somce selections prefened. Criteria for choosing the teclmologies m·e many. One key factor is the compatibility required of the many teclmologies that must be able to work together.

There will be three main environments in the CTT&DM system infrastmctme; they are the development, preproduction, and production environments. Preproduction and production

environments are VistA environments. Many servers are identified within the target technical architecture including those listed below:

• Tetminology server

• Tetminology deployment server

• VUID server

• Application server

• Continuous integration server

• Version control server

• Issue tracker server

• SQA test server

• Master File server

• Process execution and business mle server

**3.3.1.**

**Process and Activity Monitoring Server System Criticality and**

**High Availability**

System Criticality and High Availability designations are detetmined by AITC. Requirements associated with system hosting availability and uptimes are within scope of AITC. Application specific Service-Level Agreements (SLAs) be defined by Veteran Affairs protocols.

Since CTT&DM is hosted in AITC, it is covered under their Disaster Recovety Plan (DRP). CTT&DM handles administrative accmmting data and does not require any separate Recovety Time Objectives (RTOs) and Recovety Point Objectives (RPOs).

Additionally, the application does not need to be paused, or altered, during the nm process in order to guarantee a consistent and usable backup.

CTT&DM will be in an operational environment hosted within AITC and administered by VA Enterprise Operations (EO). AITC/EO have operational responsibility for the production systems it hosts, and so, is responsible for developing and providing Disaster Recover (DR)/ Continuity

of Operations Plan (COOP) policies and procedures. Additionally, there are some requirements within the SysAdmin Module related to data retention and auditing. The requirements pattially address the data recovety requirements; however AITC/EO will retain responsibility for development and providing system backup procedures and services per the existing requirements of the CTT&DM system.

**3.3.2. Special Technology**

The following table lists the special technology requirements.

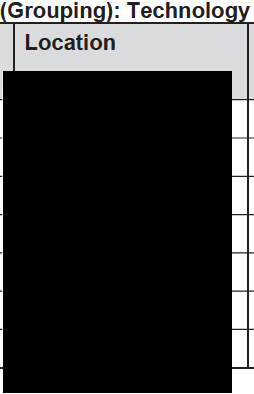
**Tabl**e **21**:**Sipec1**.**aITechno ogy ReqUirements**

|  |  |  |  |
| --- | --- | --- | --- |
| **Special Technology** | **Description** | **Notional Location** | **TRM Status** |
| N/A | N/A | N/A | N/A |

**3.3.3. Technology Locations**

The following table shows all of the locations of the cunent VETS backbone architecture and core usage.

Table 22 Location Details



Technology Component

Production

Usage

Workstations 12 X 5

Special Hardware 12 X 5

Interface Processors 12 X 5

Legacy Mainframe 12 X 5

Legacy Application Server 12 X 5

Legacy Databases 12 X 5

Other 12 X 5

|  |  |  |
| --- | --- | --- |
| Technology Component  Production 2 | Location | Usage |
| TBD | TBD | TBD |

|  |  |  |
| --- | --- | --- |
| Technology Component  Certification | Location | Usage |
| TBD | TBD | TBD |

|  |  |  |
| --- | --- | --- |
| Technology Component  Education | Location | Usage |
| TBD | TBD | TBD |

|  |  |  |
| --- | --- | --- |
| Technology Component  Test | Location | Usage |
| TBD | TBD | TBD |

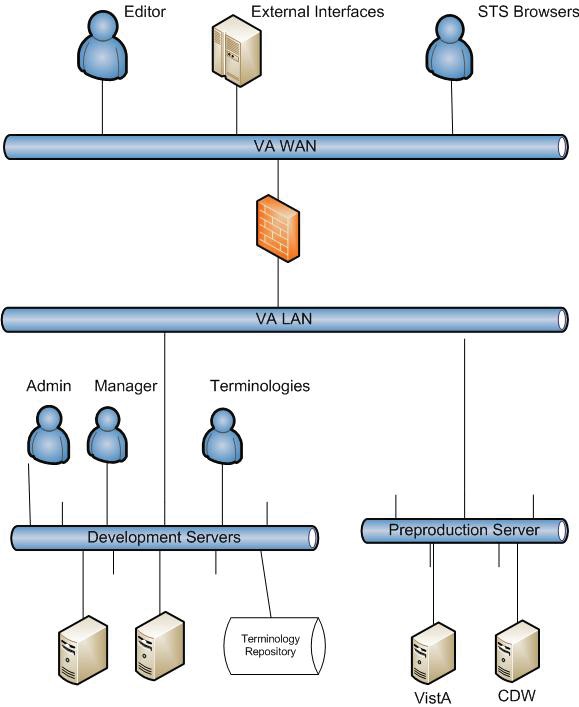
|  |  |  |
| --- | --- | --- |
| Technology Component  Development | Location | Usage |
| TBD | TBD | TBD |

**3.3.4. Conceptual Infrastructure Diagram (Server)**

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

The figure below shows the configuration of the CTT&DM system and depicts the location of the environments and the external interfaces. There are two environments, development and preproduction, along with the VistA, Software Quality Assurance (SQA) test environments. Both are clustered and distributed environments. The terminology repository is also made accessible

to the public, through a web-based application server for editors and programmable access.



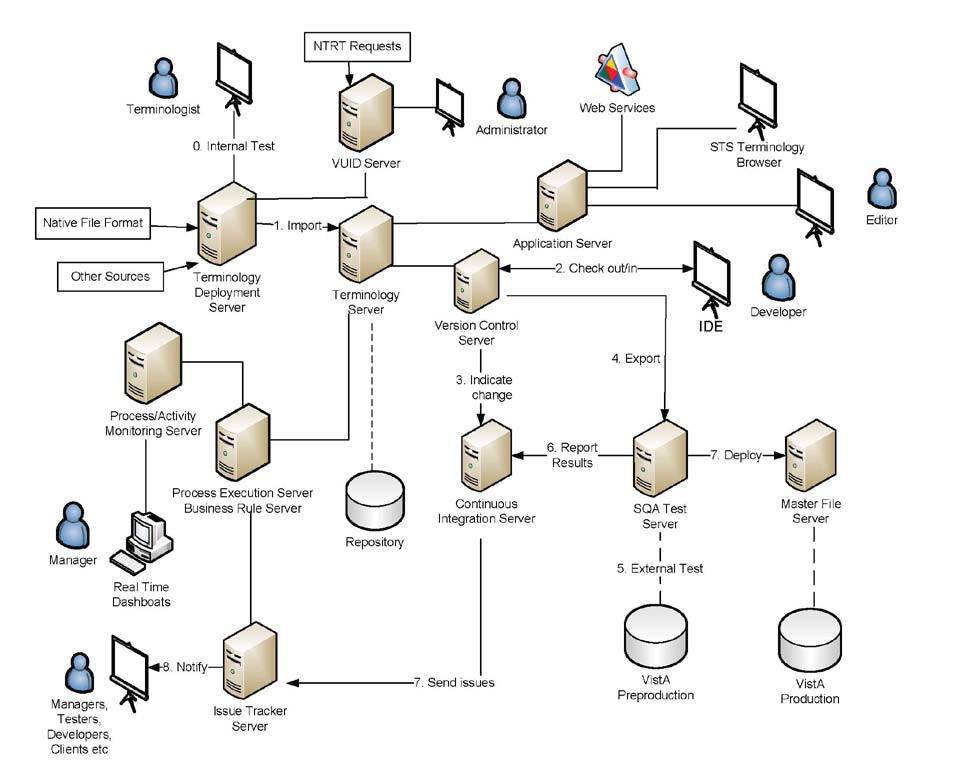
**Figure 9: CTT&DM Conceptual Networks and Environments**

**3.3.4.2. Conceptual Production String Diagram**

The figure below depicts the server architecture that hosts the CTT&DM system. It is a “To-Be” server architecture that integrates three subsystems: VETS, Continuous Integration (CI), and Business Process Management (BPM). Each subsystem has its own architecture. The integration of the three architectures should be seamless and interactive as a whole system.

VETS receives data inputs from three sources which are then converted to XML. Some of the data inputs are internally tested and some are imported to the repository of the terminology server. CI then coordinates the 8-step publication process including, Version Control and Issue Tracker. BMP tracks/monitors the projects status and assigns tasks to users.

**Figure 10: Conceptual Production String Diagram**



**4. System Architecture**

The following sections outline the conceptual model that defines the structure, behavior, and many views of the CTT&DM system. The conceptual model makes use of elements of both software and hardware, and it is used to enable design of such a composite system. It is a logical

‘partitioning scheme’ which naturally divides all of the present and foreseeable requirements of the system into a workable set of cleanly bounded subsystems with nothing left over. Also, it describes all of the necessary, or core, subsystems. The system architecture is platform- independent and captures all business, technology, and contextual interaction aspects at a conceptual level.

**4.1. Hardware Architecture (Server)**

The suggested hardware architecture for the CTT&DM system is both distributed and centralized computing. It is distributed in development and preproduction or user acceptance testing. It is centralized in repository storage and centralized backup. As the number of users of the

CTT&DM system increases, the system can scale out to add more servers as needed. The benefits include, but are not limited to, the following:

1. Lower capital and operational costs upfront (minimal hardware investment upfront).

2. Security (all data is stored within a secured datacenter).

3. Less administrative costs/overhead (fewer resources are needed since all equipment is in one location).

4. Less backup complexity.

5. Greater control over potential risk areas, such as, Internet access.

6. Each scaled out server survives on its own, so there is no Single Point of Failure in this regard.

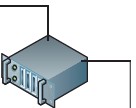
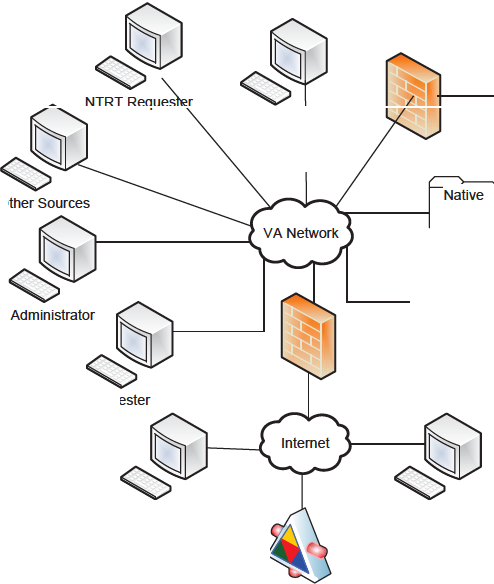
The basic architecture of the distributed and centralized computing is N-tier architecture. N-Tier typically refers to web applications that further forward their requests to other enterprise services. This type of application is the one most responsible for the success of application servers. Another basic aspect of distributed computing architecture is the method of communicating and coordinating work among concurrent processes. Through various message passing protocols, processes may communicate directly with one another, typically in a master/slave relationship. Alternatively, a "database-centric" architecture can enable distributed computing to be done without any form of direct inter-process communication by utilizing a shared database.

The hardware architecture for the CTT&DM system is depicted in the figure below. There are two development servers and four preproduction test servers. The number of both development and test servers is not fixed; instead, the number of these servers may be increased, as needed, to meet expanding business requirements.

Austin Information Technology Center

Distributed archtiecture

Centrailzed repostiory and backup



Terminology Development Server Cluster

SQA Test Server

Cluster

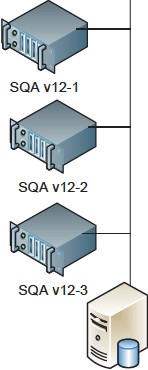
Terminal gy

Develop r

Other Sources

Files SQA V11.0.1

DEV V12-1



v-

<G0 D

Manager

Tester

DEV V12-2

Terminology

Viewers

Terminology

Edrtors

DB Server01

Reposrtory



Web Service

Cilents

DB Server02

Backup

DB Server03

Test Data

Figure 11: Hardware Architecture Diagram

Potential teclmologies to facilitate the distributed computing and centralized storage model are listed below:

• VMware- Virtual Desktop Teclmology.

• Citrix XenApp- F01merly Presentation Server, a long-time leader in the Application

Delive1y market.

• Microsoft Terminal Server- Often refened to as 'Citrix-light'; it has made great improvements in Windows 2008.

• Riverbed - This market leader in wide area network (WAN) optimization has several different models for small and midsize offices. For an added 'cool' factor, some of their devices come embedded with the free version of VMware, allowing the purchaser to deploy up to 5 Virtual Machines, nmning inside the Riverbed appliance.

• Hosted Backups - There are many vendors in this space; utilizing them for back-up data at remote sites may eliminate this hurdle for distributed computing.

**4.2. Software Architecture (Tooling)**

The default platform for all software development beneath the presentation layer development will be Oracle 64 bit Java 8 SE. The default framework for all web application presentation layer development will be done in HTML5 markup, with JavaScript as the Document Object Model (DOM) scripting language, and with support for high-resolution display devices. Scaled Vector Graphic (SVG) images are the preferred means for scalable image representations to support

high-resolution display devices. Alternatively, when SVG imagery is not appropriate, developers may use the image-set () CSS capability for raster images. The default framework for all Java Rich Internet Application (RIA) development will be JavaFX 8. Representational State Transfer (REST) is the default architectural style for providing web services. Apache Jersey 2.6 is the default framework for implementing web services. All project artifacts shall be managed by an Apache Maven compatible repository manager, and will have an associated Project Object

Model (POM) file that contains all the elements required for deployment of artifacts to the Maven central repository. For this specification, an artifact is a resource used by—or produced from—the development process, whether it is documentation, data, or an executable file. Additional existing software development tools should be directly applied where possible - make use of unit test frameworks (JUnit), defect reporting systems (Jira), version-control systems

(Git), configuration management systems (Maven), and continuous integration environments

(Hudson).

The figure below portrays the software architecture.

**Software Architecture v.1.0**

Standalone Terminology IDE Client Server Terminology IDE Terminology Web Application Environment

New Concept Request Web Application Environment



**BAM I STS Web App**

s

Presentation Layer

• User roles

• Ul components

• Ul process components

Business Process Layer

• End-to-End Business

Processes

• Business Activity

Monitoring

• External Terminology

Viewing and Editing

Composite Services

I

Integration

I Services

Service Layer

Task Web Services

I HumaTask II Technical II Data I Business

• Integration Web

Serv1ces Services Services

I **Web Service Platform**

Services

I

Services

• Data Web Services

• Web Service Platform

Service

Deck

Issuer

Tracker

Version Terminology SQA Test

Control Server Server

Workflow Continuous

Management Integration

Application Layer

• Packaged Applications

• Custom/Legacy App's

• User/Role/Access Info

Business Data Layer

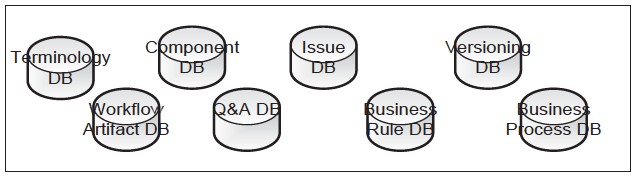
• Terminology Repository

• Project Data Store

• Application Data Stores

• Reporting Data Store

----------;:::========:;--------- Technology Layer



JAVA/J2EE I • Application Platforms

--------, • Technology Platforms

I B Server

I I Devel opment Server I 'I

Test Server • Operating Systems

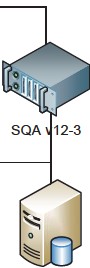
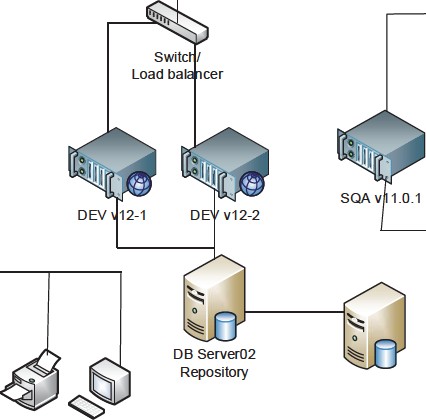
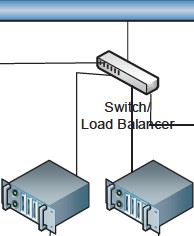
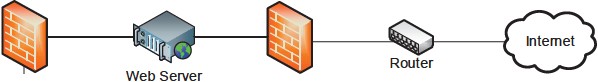
**Figure 12: Software Architecture**

**4.3. Network Architecture (Server)**

The following figure, Network Architecture Diagram specifies the physical components of the network and depicts their ftmctional organization and configmation. Both development servers and SQA testing or preproduction servers are configmed to be distributed within the VA network. They are N-Tier web applications to serve both VA organizational users and extemal organizational users.

The network architectme characterizes distributed computing where the patticipating nodes hold applications of many kinds that can be deployed in different locations of the VA network. The number of development and preproduction servers and their detailed specifications, are not predetermined, but can be scaled out to maintain required capacity and perfonnance, to satisfy the operational principles and procedmes.

Routine operations include: content transfer between the development reposit01y server and the backup database server and between the reposit01y server and preproduction servers.



InternalFirewall External Firewall

Localfvea Network

SQA 12-2 SQA v 2-1

DB Server01



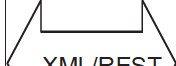
Backup

**Figure** 13: **Network Architecture Diagram**

DB Server03

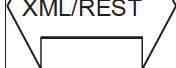
Test Data

**4.4. Service Oriented Architecture/ESS (Tooling)**



The following figure shows the Service Oriented Architecture.

**Service Oriented Architecture**



There will be two enterprise shared services which will be providing:

• Workflow Management

• WS-Human Task

PRISME will provide a workflow management service that is based on the BPMN 2.0 specifications and supports the entire life cycle of the business process (from authoring through execution, to monitoring, and management). Java Business Process Model GBPM) is an open­ source (Apache 2) prototypical example of such a workflow management service.

The Workflow Management Service shall provide Web Services-Human Task (WS-Human Task) management capabilities that include the ability to define the type of task, the priority of the task, the data associated with the task, and the ability to assign a task to an individual or a group.

There may be web services needed in order to integrate BPM, Issue Tracker, and Continuous

Integration. When the services are decided, these will be the consumed web services.

**4.5. Enterprise Architecture (Tooling)**

CTT&DM shall be in full compliance with the Product Development (PD) software engineering One-VA Technical Reference Model (TRM). In the table below are the list of One-VA TRM approved, and/or approved w/ constraints, decisions for technology and standards.

**Table** 23: **CTT&DM TRM- Technology**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Technology** | **Decision** | **Constraints (if Applicable) topproved elease**  **Date** | | |
|  | |  |
| J2EE Server | Approved | Technology must remain patched and operated in accordance with Fedral and Department security policies and guidelines in order to mitigate known and future security vulnerabilities. | 11/30/2014 | f4.1.X |
| Oracle | Approved | Technology must remain patched and operated in accordance with Fedral and Department security policies and guidelines in order to mitigate known and future security vulnerabilities. | 3/12/2015 | Current A ersion |
| Jenkins | Approved | Technology must remain patched and operated in accordance with Fedral and Department security policies and guidelines in order to mitigate known and future security vulnerabilities. | 1/20/2015 | 1.597 |
| Sonatype  Nexus | Approved | Technology must remain patched and operated in accordance with Fedral and Department security policies and guidelines in order to mitigate known and | 6.02.2015 | 2.11 |

Date

Table 24· CTT&DM TRM- Standards

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Technology | Decision | Constraints (if Applicable) tnpproved elease | | |
|  | |  |
|  |  | future security vulnerabilities. |  |  |
| jBPM | Approved | Technology must remain patched and operated in accordance with Fedral and Department security policies and guidelines in order to mitigate known and future security vulnerabilities. | 1/23/2015 | 6.1 |
| Sonarqube | Approved | Technology must remain patched and operated in accordance with Fedral and Department security policies and guidelines in order to mitigate known and future security vulnerabilities. | /2/2015 | 5.1 |
| jQuery | Approved | Technology must remain patched and operated in accordance with Fedral and Department security policies and guidelines in order to mitigate known and future security vulnerabilities. | 5/12/2015 | 2.1.X |
| GIT | Unapproved | (Requesting a TRM update or waiver.) |  |  |
| JIRA | Unapproved | (Requesting an TRM update or waiver.) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Standard | Decision | Constraints (if Applicable) Approved Release | | |
| Date | |  |
|  |  |
| HTML5 | Approved w/Constraints  Planning/Evaluation  Constraint | 1. Use of HTML 5 is limited to internal VA websites for evaluation use only and must continue to be  508 accessible and produce error- free functionality when using VA common desktop configuration. The HTML 4.01 version will remain the official version of HTML supported in VA at this time.  2. This entry is currently being evaluated, reviewed, and tested in controlled environments. Use of this technology is strictly controlled and | 12/31/2013 | 5.0CR |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Standard** | **Decision** | **Constraints (if Applicable) Approved Release** | | |
| **Date** | |  |
|  |  |
|  |  | not available for use within the general population. |  |  |
| Ruby | Approved w/Constraints | Projects using this technology should employ a Test-Driven development model, and allow sufficient time to  the functional testing phase to ensure adequate testing of dynamic typing. | /4/2014 | .1.0 |
| Java SE | Approved |  | 11/30/2014 |  |
| Scalable Vector Graphics (SVG) | Approved |  | *p!30I2015* | 1.1 nd  Edition |
| Xquery | Approved |  | /29/2015 | p.o |
| JavaScript | Approved |  | 10/31/2014 | Edition  .1 |
| Java EE | Approved |  | 10/31/2014 |  |
| OWL2 | Approved |  | /23/2014 |  |
| HL7 | Approved w/Constraints | Please contact Veterans Health Administration (VHA), Chief Health Informatics Officer (CHIO) Standards  & lnteroperability prior to implementation to obtain guidance and the VHA-endorsed version of this standard.  This entry is currently being evaluated, reviewed, and tested in controlled environments. Use of this technology is strictly controlled and not available for use within the general population. |  |  |
| BPMN | Approved |  | /27/2014 | .0 |
| Logical | Approved | Please contact VHA, CHIO | /17/2014 | .4 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Standard** | **Decision** | **Constraints (if Applicable) Approved Release** | | |
| **Date** | |  |
|  |  |
| Observation Identifier Names and Codes (LOINC) | w/Constraints | Standards & lnteroperability prior to implementation to obtain guidance and the VHA-endorsed version of this standard. |  |  |
| Systematized Nomenclature of Medicine - Clinical Terminology (SNOMED - CT) | Approved w/Constraints | Please contact the VHA Office of Standards & lnteroperability (S&I) (Mail group: VHA S&l Response Team) prior to implementation to obtain guidance and the VHA- endorsed version of this standard. | /7/2015 | anuary  014 |
| Unix | Approved |  | p!3112014 | p |
| css | Approved |  | 10/31/2014 | .1 |

**5. Data Design**

This section defmes the overall data and data management systems. CTT&DM is a system of systems, many of which have their own data management systems. This section identifies the data between systems and databases that the systems will operate on. To identify the data and database management systems, we apply dataflow diagram (DFD) techniques to defme major data and their flows as well as ftmctions processing the data.

The DFD Level 0, shown in Figmel4, and the DFD Level l, shown in the following figmes, display a graphical representation of the "flow" of data through the CTT&DM system from its process aspects. These figmes show what kind of infonnation will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It visualizes the data processing from which we identify data stores. Eight data stores are identified as eight databases.

The DFD is derived from the data CTT&DM will need to process, where they are coming from, and where the processed data will flow into. The DFD proposes what ftmctions are needed in order to process the data as required. The data aspect of the DFD is derived from the Tooling BRD while the ftmctional aspect is derived from Section 2.2, Overview of the Business Process, and shows the ftmctional business model. We use fom ftmctional groups as elements of the Level

0 DFD and use the thilieen identified ftmctions as the elements of the Levell DFD. The Level2

DFD is necessaty, but we limited om scope of this project at the Levell DFD.

The DFDs are condensed representations of ftmctions and data and their interaction with

CTT&DM. The CTT&DM DFD merely represents inf01mation contained in the Tooling BRD

into a new f01m understandable by development staff. The information in the Tooling BRD is not assumed to be complete, but it does contain sufficient inf01mation to diagram data flows. Therefore, the understanding of the DFD Level 0 and Level 1 requires a necessaty lmderstanding of the Tooling BRD and the statements of the problem. The DFD merely represents the problem in a new fotm. The prerequisites of the DFD audience are to have a clear understanding of the problem, which requires reading the Tooling BRD. Without a clear understanding of the BRD, it will be difficult to understand the DFD.

The pmpose of the DFD is to represent the problem into technical drawings so that development can proceed. The drawings condense large amounts of the problem inf01mation into pictmes. To accomplish this, it applies significant abstraction skill to filter out the detailed inf01mation to captm·e the whole pictm·e. The pmpose of the DFD is not to educate readers as to what the problem is. Readers of the DFD are assumed to ah·eady have ample understanding of the problem, that is, the readers are assumed to have ah·eady read and understood the Tooling BRD and RSD.

The DFD of CTT&DM represents the problem rather than a solution in three dimensions: 1) data somce, sink, and flows; 2) system and subsystems' boundaries to differentiate what are parts of the system from those that are not; 3) functions at different levels necessaty to process the data. We will provide a brief overview below of five Level-0 and Level-l diagrams by giving detailed explanation on the ftmctions and data in each DFD diagram.

**CTT&DM DFD Level 0**

The CTT&DM is split into fom functional groups: project management, tetminology development, support, and publication according to Section 2.2, Overview of the Business Process. The fom groups are the fom functions at Level 0 which defmes the b01mdaty of the CTT&DM system. Functions outside of the fom functional groups are outside of the CTT&DM boundaty. The fom functional groups entail all business ftmctional requirements documented in the Tooling BRD. The table below describes the allocation of business functional requirements to the fom groups.

**Table 25 FunctlonaIGroupmgs**

|  |  |
| --- | --- |
| **Function** | **Definition** |
| **Project**  **Management** | 5.1.1 Workflow Management Service  5.1.2 Identity and Access Management Service  5.1.4 Tracker Service  5.1.9 Project Creation Application  5.2.22 Workflow |
| **Terminology**  **Development** | 5.1.6 Version Control Service  5.1.8 Continuous Integration Service  5.2.1 Status Time Author Module Path (STAMP) Versioning Component  Database  5.2.2 Taxonomy Viewing  5.2.4-8 Query Capabilities; Reference set Capabilities; Document Generation  Capabilities; Scripting Capabilities; Description Logic  5.2.11 Template-Based Modeling |

|  |  |
| --- | --- |
| **Function** | **Definition** |
|  | 5.2.14-191nterview Driven Modeling; Steal Pieces and Re-use in a New Model;  Rules Driven Modeling  5.2.21 Context-sensitive help  5.2.24-25 Additional Considerations; Server Requirements  5.5.4 XML Transforms |
| **Support** | 5.1.3 Artifact Repository Service  5.1.5 Service Desk  5.1.7 Quality Management  5.2.3 Component Request Service  5.2.9. Mapping Capabilities  5.2.10 Browse a Set of Approved, Suggested, or Similar Models  5.2.12-13 Modify Existing; Specialize From Existing;  5.2.20 Human consults  5.2.23 Quality Assurance  5.3.4 New Concept Request Application |
| **Release and**  **Publication** | 5.5.1 Terminology Releases from Standards Development Organizations  (SDO)  5.5.2 Terminology Releases from Knowledge Sharing Partners  5.5.3 Clinical Decision Support (CDS) and Knowledge Related Artifacts  5.6 Outputs from System  5.6.1 Publishing Output Content (VETS) |

There are three kinds of data entering and leaving the CTT&DM system. They are listed below:

1. Tenninology data that is fmther broken down into three kinds:

• VA's intemal tenninology standard (VHAT);

• Map Sets, which map between concepts in different terminologies; and

• Standard Code Systems (SCS) which are tenninologies, developed by Standards

Development Organizations, such as SNOMED CT, ICD-10, and LOINC.

2. Project mtifacts that m·e published and made accessible to other organizations and project mtifacts made by other organizations are made accessible to CTT&DM.

3. Components, new tenns added to one of the SCS Systems, new CDS or Knowledge Artifacts, new automation mles for scripting the behavior of an application, new mles for confnming the quality or conectness of an editing action, or batch quality assurance and related components managed as pa1t of the ETS. The components may be from extemal standard organizations such as SNOMED.

There are three kinds of extemal entities that are data sources and data sinks.

1. Extemal consumers including SDO, the NLM US realm/NLM, Healthcare and Health IT

pmtners and other authoritative sources.

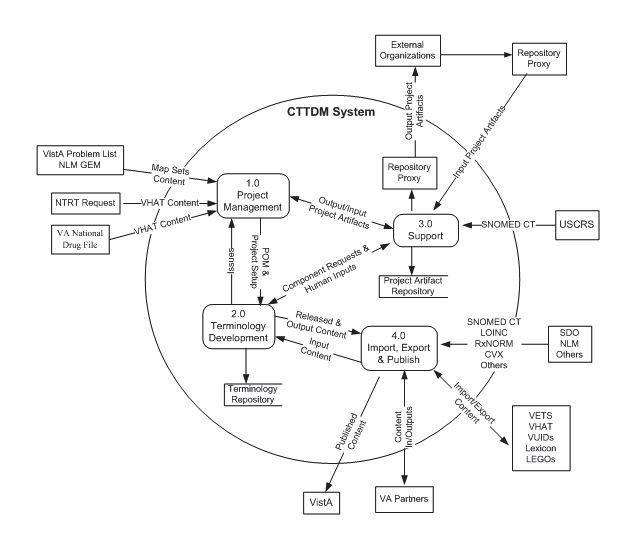
2. Knowledge sharing pmtners that create and publish tetminologies.

3. VistA organizations that create NTRR requests and receive terminology releases

In the Level 0 DFD diagram, the starting point is VistA organizations entering VA terminology requests and Map Set requests into project management function, that create projects for terminology development. The developed new terminologies are then released, or published, into all three kinds of external entities.

Project artifacts are produced and acquired by external organizations, such as knowledge sharing partners. Project artifacts created by knowledge sharing partners, flow into the system as needed. Components enter into the system upon request and Virtual User Input Devices (VUIDs) are assigned through the function of Import, Export, and Publish.

DFD diagrams at Level 1 are self-explanatory once the Level 0 diagram is understood. We briefly describe DFD diagrams at Level 1.

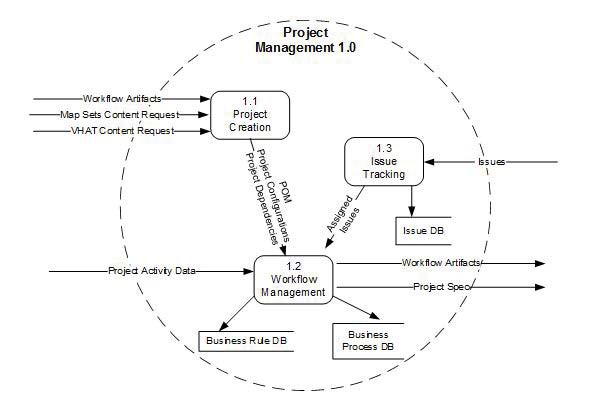


**Figure 15: CTT&DM DFD Level 0**

**Project Management 1.0**

This function contains three functions at the next level: project creation, workflow management, and issue tracking. There are five data inputs and two data outputs consistent with the Level 0

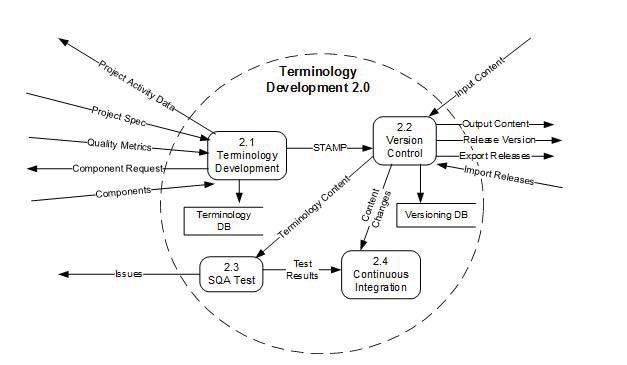
DFD. There are three data stores in this function: issue DB, business rule DB, and business process DB.



**Figure 16: CTT&DM Project Management Level 1 DFD**

**Terminology Development 2.0**

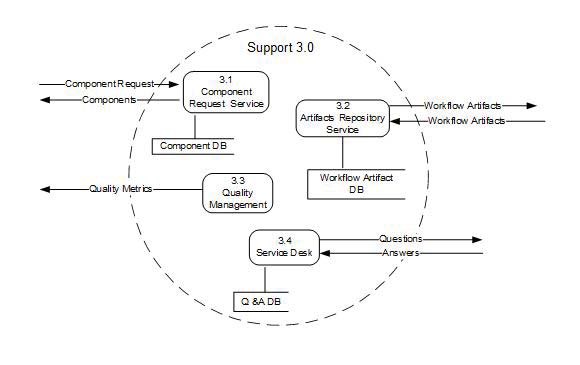
This function contains four functions at the next level and two data stores. The four functions are new concept creation, version control, continuous integration, and SQA test. The two data stores are terminology database (DB) and versioning DB. Data inputs and exports are consistent with the Level 0 DFD.



**Figure 17: CTT&DM Terminology Development Level 1 DFD**

**Support 3.0**

This function contains four functions at the next level and three data stores. The four functions are the component request service, artifacts repository service, quality management, and service desk. The three data stores are component, project artifacts, and Q&A databases. Data coming and leaving the function are consistent with the Level 0 DFD.



**Figure 18: CTT&DM Support Level 1 DFD**

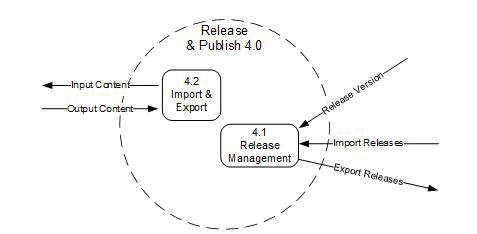
**Release and Publish 4.0**

This function contains two functions at the next level: import & export and release & publish. Data coming in and leaving the function are consistent with the DFD Level 0.

In summary, data flow diagrams are process models partially describing the system that exists and the system that is required. The model shows what is currently being done with the information that is entering the system. The process model does not show how the information is organized but, act as a focus for discussion between the developer and the user, to determine

what the new system should be able to do. The models will also provide a measure during testing to see if the system produced does do what was promised. Typically, the processes shown on a DFD will in some way translate into screen designs in the final system. For example, a process

on a DFD may evolve as a menu item that when clicked, opens in a window to allow data input, output, or simply review.



**Figure 19: Release and Publish Level 1 DFD**

**5.1. Data Base Management System (DBMS) Files**

Some databases are embedded in commercial-off-the-shelf (COTS) systems, and others are used for different things. The following database servers have their own database management capabilities: Issue tracking system, continuous integration server, and JIRA. Hudson, Jenkins, and Bamboo all have their own data repository. Bamboo is interfaced with a JBDC driver, and can hook up with SQL. Artifact repositories use a hybrid approach, have multiple ways to manage a system, and interface through Java database connectivity (JDBC). The Mongo database is used for mobile applications. The Version Control Service shall have the necessary storage, configuration, and bandwidth to support version control of large textual data sources, such as a complete history of all Systematized Nomenclature of Medicine (SNOMED) release files (Systematized Nomenclature of Medicine Clinical Terms [SNOMED CT] is released biannually, and the data files of the release are ~1.3 GB, compressible to ~200 MB).

The capability to update versions will come in due time. Every object has STAMP versioning, but there are not normalized tables for the logical design. DBMS schemas, subschemas, records, sets, tables, and storage page sizes come from the relationship records. The access methods are indexed, sequentially, and random access. The estimated database file size including overheard resulting from access methods and free space is 2-5 GB. The estimation on the number of transactions that the database may have to process is as follows

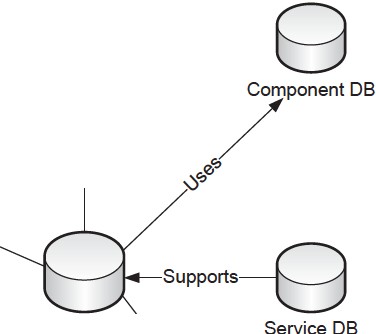
x In the editing environment, there are five commits per author, per hour, which comes to a commit every four minutes.

x The run time environment will need to be discussed more and is to be determined (TBD).

**5.2. Data View**

CTT&DM is a distributed system integrating several systems. These systems are either embedded or SaaS systems integrated through SOA architecture. Many of these systems have their own databases. Below are eight databases and their relationships identified for the CTT&DM system.

Terminology D"B



'*' 'Oc:Jllc*

Workflow Artifact DB

8

**t**

Ill Q) Ill

::::>

Ill

e

**t**

c:

0

I

8

Versioning DB

*es*

Business

Process DB

**t**

Icll

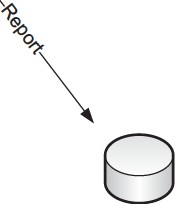
-

Cii

c

0

8



Issue DB

Business Rule DB

**6. Detailed Design**

**Figure** 20: **Data View**

The system shall be developed using an open source, non-Windows-dependent architecture. The system servers shall employ servers that include, but are not limited to: an Application Server, Continuous Terminology Server, Database Server, and a Versioning Control Database Server.

The application server is a platf01m for developing and deploying multitier distributed Enterprise Collaborative Te1minology Tooling that will be centralized on an integrated server which will utilize technologies, such as caching and connection pooling, to improve resource use and application perf01mance, as well as other components/tools for improved perf01mance. The application server shall provide the following features/functionality:

• Shall be developed on a J2EE Platf01m.

• Shall be developed on a Non-Windows Dependent OS.

• Shall be developed using XML technologies. XML describes the content and structure of data in a document and is an industry standard for delivering content on the Intemet. Typically, XML is used as the data exchange f01mat between J2EE applications and client applications, or between components of a J2EE application.

• Server shall support Web browsers and other clients that use HTTP/HTTPS.

x Server shall support most commonly used Web browsers; such as Windows, Chrome, and

Mozilla Firefox, as well as mobile devices such as IOS and Android.

x Server provides a platform for deploying Web services as components of a heterogeneous distributed application. Web services use a cross-platform, cross-language data model (XML), to provide interoperability among application components on diverse hardware and software platforms.

x The web application components of the system shall support 5,000 registered users, and

600 concurrent users.

x Server shall provide Enterprise scalability.

x Servers shall have a minimum capacity of 1.2 TB, 10-15K rpm or Solid State Disk (SSD), 24 GB RAM (48 GB recommended). The servers shall have a minimum of six CPU cores available for concurrent processing with 12 CPU cores preferred; Intel Xeon E5, 3.0 GHZ, six core processors is a prototypical example, equivalent or better performance to prototypical example, would be acceptable.

**6.1. Hardware Detailed Design**

This section provides the hardware design specifications for the development and the user acceptance testing (UAT) environments. There are two development servers and two database servers for the development environment. There are four SQA test servers and one database server for the user acceptance testing environment. In the table below, the specifications for each of these nine machines are shown.

Table 26: Hardware Configuration Parameters for the Development and the UAT Environments

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Environment | Function/ Application/ Operating System  (OS) | Data  Center | vCPUs | vCOREs | RAM  (GB) | Disk OS/ Data  (GB) |
| SQA Test  Server  (UAT) | VMWare Platform  Weblogic 10.2.5  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| SQA Test  Server  (UAT) | VMWare Platform  Weblogic 10.2.5  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| SQA Test  Server  (UAT) | VMWare Platform  Weblogic 10.2.5  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| SQA Test  Server  (UAT) | VMWare Platform  Weblogic 10.2.5  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| Database  Server  (UAT) | VMWare Platform  Oracle 11g  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| Subtotal |  | | | | 80 | 2,500 |
| Development  Server | VMWare Platform  Weblogic 10.2.5  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| Development  Server | VMWare Platform  Weblogic 10.2.5  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| Database  Server  (Dev) | VMWare Platform  Oracle 11g  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| Database  Server  (Dev) | VMWare Platform  Oracle 11g  RHEL 6.2 | AlTC | 1 | 2 | 16 | 100/  400 |
| Subtotal |  | | | | 64 | 2,000 |
| Grand Total |  | | | | 192 | 4,500 |

Note that the disk storage specifies separate logical drives designated for use by the operating system and for system application data. The hardware configurations assume the use of virtual machines (VMs).

**Table** 27: **Production System Components**

|  |  |
| --- | --- |
| **System**  **Component Query** | **Description** |
| Racks | Racks NOT YET SPECIFIED. Note: The Linux servers mentioned in the table above are Virtual machines and running on an ESX VMware Farm. |

**6.2. Software Detailed Design**

Details will be defined during the development cycle and are not applicable at this time.

**6.2.1. Conceptual Design**

The Conceptual Design for CTT&DM is defined above in Section 3.1, Conceptual Application

Design.

Details will be defined for the Server during the development cycle and are not applicable at this time.

**6.2.1.1. Product Perspective**

The sections below will be explained in Section 6, Software Detailed Design.

Details will be defined for the Server during the development cycle and are not applicable at this time.

**6.2.1.1.1. User Interfaces**

Details will be defined for the Server during the development cycle and are not applicable at this time.

**6.2.1.1.2. Hardware Interfaces**

No special hardware devices are used within the application. The application can be accessed via standard web browsers.

**6.2.1.1.3. Software Interfaces**

All software interfaces are detailed in the CTT&DM RSD.

**6.2.1.1.4. Communications Interfaces**

System will be accessed both intemally, within the VA, and via the public Intemet. The system will supp01i use of laptops and desktops. The users will be able to interact with the system via a web browser.

**6.2.1.1.5. Memory Constraints**

The system will be accessed both internally, within the VA, and via the public Internet. The system will support use of laptops and desktops. The users will be able to interact with the system via a web browser.

**6.2.1.1.6. Special Operations**

The only special operations required by the user are operations such as backup, recovery, and archiving. There won’t be any use of external storage devices with the system. There won’t be any additional hardening of the applications installed locally on the system.

**6.2.1.2. Product Features**

The product features are better described in the user manual.

**6.2.1.3. User Characteristics**

Users of the CTT&DM system server-side, are current users of the VETS application and will need any additional training, as necessary, when the system is implemented into the environment.

The CTT&DM system consists of users who perform the following tasks:

x The system integrates with VA business applications (as determined to be feasible) across heterogeneous environment and platforms.

x Author content

x Review and update functions performed by Domain Reviewers

x Deployment functions performed by Testing Coordinators and other authorized

Deployers

**6.2.1.4. Dependencies and Constraints**

The following is a list of dependencies and constraints for the CTT&DM system.

x Regulatory policies

x Must run on UNIX OS and be able to run in concert with existing system components.

x The default platform for all software development beneath the presentation layer will be

Oracle 64 bit Java 8 SE.

x Application is to be built on open-source architecture.

x Must be accessible over the public Internet.

x PRISME will provide a workflow management service that is based on the **Business**

**Process Model and Notation (BPMN) 2.0.**

x The workflow management service engine must be able to run embedded within the default Java 8 Standard Edition (Java 8 SE) application platform.

x The default framework for all web application presentation layer development will be done in HTML5 markup, with JavaScript as the Document Object Model (DOM)

scripting language, and with support for high-resolution display devices. SVG images are the preferred means for scalable image representations to support high-resolution display devices. Alternatively, when SVG imagery is not appropriate, developers may use the image-set () CSS capability for raster images.

x The default framework for all Java Rich Internet Application (RIA) development will be

JavaFX 8.

x REST is the default architectural style for providing web services.

x Apache Jersey 2.6 is the default framework for implementing web services.

x The Workflow Management Service shall publish workflow processes as Maven or

Maven plug-in artifacts.

x The Version Control Service shall have storage, configuration, and bandwidth to support version control of large textual data sources, such as a complete history of all SNOMED release files (SNOMED CT is released biannually, and the data files of the release are

~1.3 GB, compressible to ~200 MB).

x The rich client tools will support scripting using the Oracle Nashorn JavaScript libraries.

x High-performance stand-alone terminology editing environment, deployed as a Java Web

Start rich internet application able to efficiently run on a laptop computer.

x The Stand-alone Terminology Integrated Development Environment must support all the capabilities currently supported by the IHTSDO and ISAAC workbench.

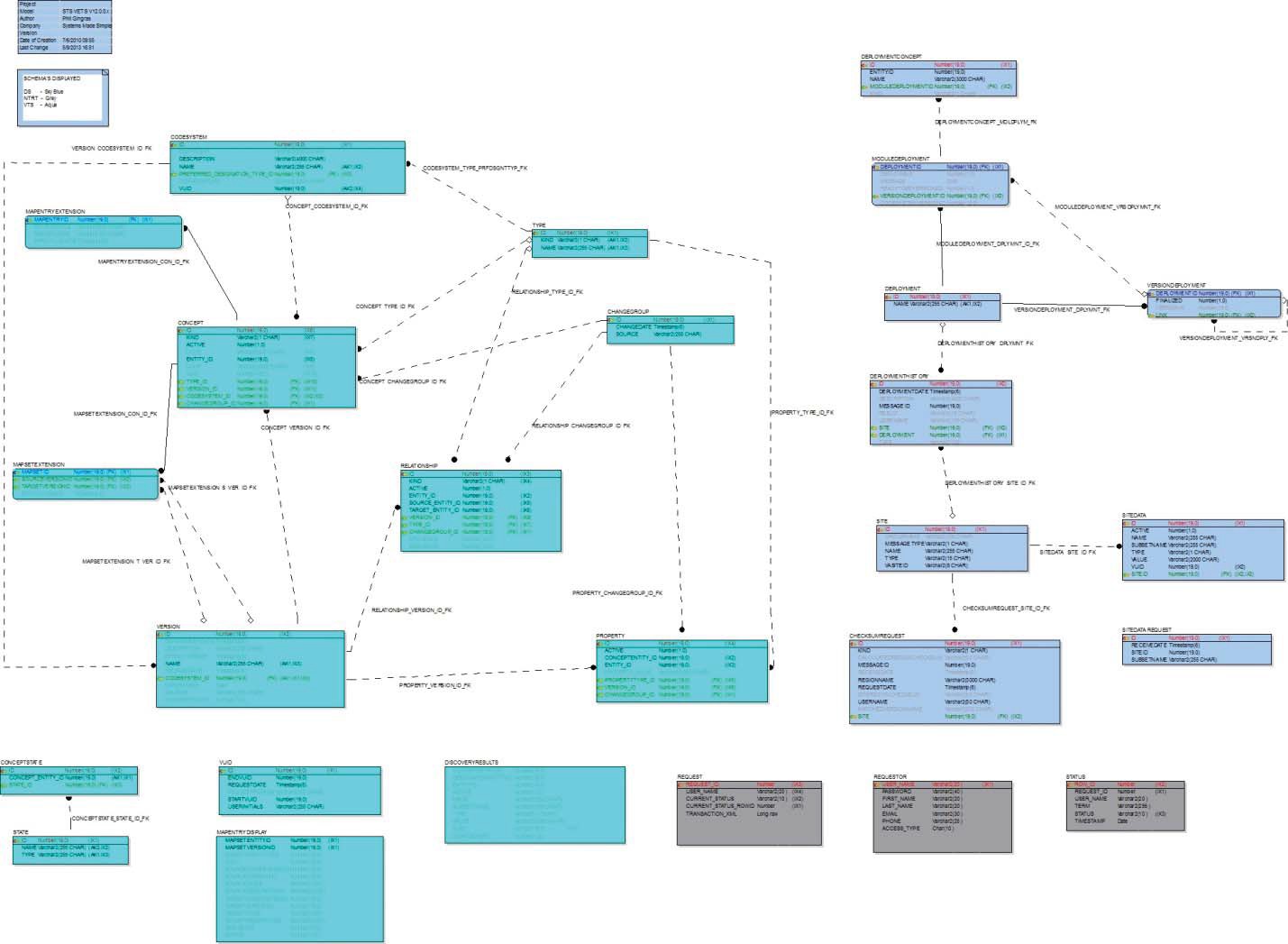
x The system will be able to accept, process, integrate, and version electronic releases from terminology knowledge sharing partners.

**6.2.2. Specific Requirements**

Specific requirements are defined in the “To-Be” Requirements. The traceability requirements are detailed above in Section 2.3, Overview of the Significant Requirements.

**6.2.2.1. Database Repository**

The database used to house the current terminology content is VETS. There is no determination as to the proposed logical database design of the CTT&DM. The following figure is a diagram of the database repository.



**Figure 21: Database Repository Diagram**

**6.2.2.2. System Features**

The proposed system features of the CTT&DM server are described in the list below:

x Terminology Server (Runtime Service) – Provides programmatic access to terminology content.

x Continuous Integration Server - Provides the ability to manage the developmental workflow by providing an automated process.

x Artifact/Project Repository – Provides the ability to apply version control and share common libraries.

x Version/ Source Control - Provides a distributed version control system, where clinical terminologists will have the ability to work on the same terminology/project at the same time. Version control will be provided for all CTT&DM content. Each file will be tracked, any modifications made, additions, deletions, or movement of files will receive a version number upon commit.

x Workflow Management and Rules Engine - Supports the business life cycle from authoring, through publication, of terminology content, to monitoring and management – utilizing a workflow management and rules engine.

• Component Request Service - Provides the ability to allow users to enter modification requests for tenninology that can be tracked and managed globally.

• Quality Management - Provides continuous inspection of content quality utilizing business mles from local validation, through release validation.

• Database Server - The server will provide storage, the ability to manipulate data, archiving, and versioning. All changes made to the database will be tracked.

**6.2.2.3. Design Element Tables**

Not Applicable - CTT&DM is not a VistA package.

**6.2.2.3.1. Routines (Entry Points)**

Not Applicable - CTT&DM is not a VistA package.

**Table** 28: **Routines (Instructions)**

|  |  |
| --- | --- |
| **Routines** | **Instructions** |
| **Routine Name** | N/A |
| **Enhancement**  **Category** | N/A |
| **RTM** | N/A |
| **Related Options** | N/A |
| **Related Routines** | N/A |
| **Data Dictionary (DO) References** | N/A |
| **Related Protocols** | N/A |
| **Related Integration Control Registrations (ICRs)** | N/A |
| **Data Passing** | N/A |
| **Input Attribute Name and Definition** | N/A |
| **Output Attribute**  **Name and Definition** | N/A |
| **Current Logic** | N/A |
| **Modified Logic (Changes are in bold)** | N/A |

**Table 29 :Routines**



**Routines Activities**

|  |  |
| --- | --- |
| Routine Name | N/A |
| Enhancement Category | DNew 0 Modify D Delete D No Change |
| RTM | N/A |
| Related Options | N/A |

|  |  |  |
| --- | --- | --- |
| Related Routines | Routines "Called By" | Routines "Called" |
|  | N/A | N/A |

|  |  |
| --- | --- |
| Routines | Activities |
| Data Dictionary (DO) References | N/A |
| Related Protocols | N/A |
| Related Integration Control Registrations (ICRs) | N/A |
| Data Passing | D Input D Output 0 Both D Global D Reference Reference Local |
| Input Attribute Name and  Definition | Name: Definition: |
| Output Attribute Name and Definition | Name: Definition: |

I Current Logic

N/A

Modified Logic (Changes are in bold)

N/A

6.2.2.3.2. Templates

Not Applicable - CTT&DM is not a VistA package

Table 30: Templates (Instructions)

|  |  |
| --- | --- |
| Templates | Instructions |
| Template Name | N/A |
| Enhancement Category | N/A |
| RSD Traceability | N/A |

|  |  |
| --- | --- |
| Templates | Instructions |
| Template Type | N/A |
| Related Options | N/A |
| Related Routines | N/A |
| Data Dictionary (DO) References | N/A |
| Global References | N/A |

Table 31 (Grouping): Templates

|  |  |
| --- | --- |
| Templates | Description |
| Template Name | N/A |
| Enhancement Category | DNew 0 Modify D Delete D No Change |
| RSD | N/A |
| Template Type | D Sort D Input D Print D Other |
| Related Options | N/A |

|  |  |  |
| --- | --- | --- |
| Related Routines | Routines "Called By" | Routines "Called" |
| N/A | N/A | N/A |

|  |  |
| --- | --- |
| Routines | Description |
| Data Dictionary (DO) References | N/A |
| Global References | N/A |

6.2.2.3.3. Bulletins

Not Applicable - CTT&DM is not a VistA package

Table 32: Bulletins (Instructions)

|  |  |
| --- | --- |
| Bulletins | Instructions |
| Bulletin Name | N/A |
| Enhancement Category | N/A |
| RTM | N/A |
| Related Options | N/A |
| Related Routines | N/A |
| Mail Subject | N/A |

|  |  |
| --- | --- |
| Mail Group | N/A |
| Parameters | N/A |
| Data Dictionary (DO) References | N/A |

Table 29 (Grouping): Bulletins

|  |  |
| --- | --- |
| Bulletins | Description |
| Bulletin Name | N/A |
| Enhancement Category | DNew D Modify D Delete D No Change |
| RTM | N/A |

|  |  |  |
| --- | --- | --- |
| Related Routines | Routines "Called By" | Routines "Called" |
| N/A | N/A | N/A |

|  |  |
| --- | --- |
| Routines | Description |
| Mail Subject | N/A |
| Mail Group | N/A |
| Parameters | N/A |
| Data Dictionary (DO) References | N/A |

6.2.2.3.4. Data Entries Affected by the Design

Not Applicable - CTT&DM is not a VistA package

Table 33· Data Entries Affected by the Design

|  |  |  |
| --- | --- | --- |
| Field Name | Current Value | New Value |
| N/A | N/A | N/A |

6.2.2.3.5. Unique Record(s)

No lmique records have been identified at this time.

Table 34 Un.lque RecordiD

|  |  |  |
| --- | --- | --- |
| Field Name(s) | Current Value | New Value |
| N/A | N/A | N/A |

6.2.2.3.6. File or Global Size Changes

Not Applicable - CTT&DM is not a VistA package

|  |  |  |
| --- | --- | --- |
| Table 35 F"1le or Glob aIs·IZe Chan | | es |
| File/Global Name(s) | Estimated Increase | Estimated Decrease |
| N/A | N/A | N/A |

6.2.2.3.7. Mail Groups

Not Applicable - CTT&DM is not a VistA package

Table 36: Mail Groups (Instructions)

|  |  |
| --- | --- |
| Mail Groups | Instructions |
| Mail Group Name | N/A |
| Enhancement Category | N/A |
| Related Options | N/A |
| Related Routines | N/A |
| Data Dictionary (DDs) References | N/A |
| Related Protocols | N/A |
| Mail Group Description | N/A |
| Self-Enrollment Allowed | N/A |
| Type | N/A |

Table 37 (Grouping): Mail Groups

|  |  |
| --- | --- |
| Mail Groups | Activities |
| Mail Group Name | N/A |
| Enhancement Category | DNew D Modify D Delete D No Change |
| Related Options | N/A |

|  |  |  |
| --- | --- | --- |
| Related Routines | Routines "Called By" | Routines "Called" |
| N/A | N/A | N/A |

|  |  |
| --- | --- |
| Mail Groups | Instructions |
| Data Dictionary (DO) References | N/A |
| Related Protocols | N/A |
| Mail Group Description | N/A |
| Self-Enrollment Allowed | D Yes DNo |
| Type | D Public DPrivate |

6.2.2.3.8. Security Keys

Secmity Keys will be developed as development proceeds.

Table 38: Security Keys (Instructions)

|  |  |
| --- | --- |
| Security Keys | Instructions |
| Security Key Name | TBD |
| Enhancement Category | TBD |
| Related Options | TBD |
| Related Routines | TBD |
| Data Passing | TBD |
| Security Key  Description | TBD |
| Subordinate Keys | TBD |
| Mutually Exclusive  Keys | TBD |
| Granting Condition  Logic | TBD |
| Current Logic | TBD |
| Modified Logic  (Changes are in bold) | TBD |
| Hierarchical  Precedence | TBD |

Table 39 (Grouping): Security Keys

|  |  |
| --- | --- |
| Security Keys | Activities |
| Security Key Name | TB |
| Enhancement  Category | DNew D Modify D Delete D No Change |
| Related Options | TB |

|  |  |  |
| --- | --- | --- |
| Related Routines | Routines "Called By" | Routines "Called" |
| **TBD** | TBD | TBD |

|  |  |
| --- | --- |
| Security Keys | Activities |
| Data Passing | D D D Both D Global D Local  Input Output Reference Reference |

|  |  |
| --- | --- |
| Security Keys | Activities |
| Security Key  Description | TBD |
| Subordinate Keys | TBD |
| Mutually Exclusive  Keys | TBD |
| Granting Condition  Logic | TBD |

I Current Logic

TBD

Modified Logic (Changes are in bold)

TBD

|  |  |
| --- | --- |
| Security Keys | Activities |
| Hierarchical  Precedence | TBD |

6.2.2.3.9. Options

All Options will be captured directly from VA FileMan DDs after the fact.

Table 40: Options (Instructions)

|  |  |
| --- | --- |
| Options | Instructions |
| Option Name  (MENU TEXT field) | TBD |
| Enhancement Category | TBD |
| Associated Menu Options that will invoke this reference | TBD |
| Data Passing | TBD |
| Menu Text Description | TBD |
| Option Type | TBD |
| Option Definition | TBD |
| Current Entry Action  Logic | TBD |

|  |  |
| --- | --- |
| Options | Instructions |
| Modified Entry Action Logic (Changes are in bold) | TBD |
| Current Exit Action  Logic | TBD |
| Modified Exit Action  Logic  (Changes are in bold) | TBD |

Table 41 (Grouping): Options

|  |  |
| --- | --- |
| Options | Activities |
| Option Name | TBD |
| Enhancement  Category | DNew 0 Modify D Delete D No Change |
| Associated Menu Options that will invoke this reference | TBD |
| Data Passing | D Input D Output 0 Both D Global D Local Reference  Reference |
| Menu Text  Description |  |
| Option Type | DEdit D Print 0Menu D Inquire  D Action D Run Routine D Other |
| Associated Routine | TBD |
| Option Definition | TBD |

I Current Entry Action Logic

TBD

Modified Entry Action Logic (Changes are in bold) TBD

I Current Exit Action Logic

TBD

Modified Exit Action Logic (Changes are in bold) TBD

6.2.2.3.10. Protocols

All protocols will be captured directly from the VA FileMan DDs after the fact.

Table 42: Protocols (Instructions)

|  |  |
| --- | --- |
| Protocols | Instructions |
| Protocol Name | TBD |
| Enhancement Category | TBD |
| Associated Protocols | TBD |
| Data Passing | TBD |
| Item Text Description | TBD |
| Protocol Type | TBD |
| Associated Routine | TBD |
| Current Entry Action  Logic | TBD |
| Modified Entry Action  Logic  (Changes are in bold) | TBD |
| Current Exit Action  Logic | TBD |
| Modified Exit Action  Logic  (Changes are in bold) | TBD |

Table 43 (Grouping): Protocols

|  |  |
| --- | --- |
| Protocols | Activities |
| Protocol Name | TBD |
| Enhancement  Category | D New D Modify D Delete D No Change |
| Associated  Protocols | TBD |
| Data Passing | D Input D Output D Both D Global Reference D Local Reference |
| Item Text  Description | TBD |
| Protocol Type | D Action D Menu D Protocol D Protocol Menu D Limited Protocol D  Extended Action D Dialog D Other |
| Associated Routine | TBD |

~~I~~ Current Entry Action Logic

TBD

Modified Entry Action Logic (Changes are in bold)

TBD

~~I~~ Current Exit Action Logic

TBD

Modified Exit Action Logic (Changes are in bold)

TBD

6.2.2.3.11. Remote Procedure Call (RPC)

Any remote procedure calls will be captured directly from the VA FileMan Data Definitions

(DDs) after the fact.

Table 44: RPCs (Instructions)

|  |  |
| --- | --- |
| RPCs | Instructions |
| Name | TBD |
| TAGARTN | TBD |
| Input Parameters | TBD |
| Results Array | TBD |
| Description | TBD |

Table 45 (Grouping): RPCs

|  |  |
| --- | --- |
| RPCs | Activities |
| Name | TBD |
| TAGARTN | TBD |
| Input Parameters | TBD |
| Results Array | D Single Value D Array D Word Processing  D Global Array D Global Instance |
| Description | TBD |

6.2.2.3.12. Constants Defined in Interface

Table 46: Constants Defined in Interface

I

I Name Description

TBD

TBD

6.2.2.3.13. Variables Defined in Interface

No variables have been identified at this time.

Table 47: Variables Defined in Interface

I I

I Name Type Description

TBD

TBD TBD

6.2.2.3.14. Types Defined in Interface

No types have been identified at this time.

Table 48 Tlypes Defmi

ed.m Interface

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| TBD | TBD | TBD |

6.2.2.3.15. GUI

Graphical User Interface (GUI) designs will be provided in Section 8.4, Navigation Hierarchy.

Table 49: GUI

I I

Unit Name Description

TBD

TBD

6.2.2.3.16. GUI Classes

No classes have been identified at this time.

Table 50: GUIClasses (Instructions)

|  |  |
| --- | --- |
| GUIClasses | Instructions |
| Class Name | N/A |
| Derived From Class | N/A |
| Purpose | N/A |

Table 51: GUIClasses

|  |  |
| --- | --- |
| GUIClasses | Instructions |
| Class Name | N/A |
| Derived From Class | N/A |
| Purpose | N/A |

6.2.2.3.17. Current Form

F01m layouts will be developed as development proceeds.

6.2.2.3.18. Modified Form

F01m layouts will be developed as development proceeds.

6.2.2.3.19. Components on Form

Table 52 Components on Form

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| TBD | TBD | TBD |

6.2.2.3.20. Events

Table 53: Events

I Name I Type I Description

TBD TBD TBD

6.2.2.3.21. Methods

Table 54: Methods

|  |  |  |
| --- | --- | --- |
| Method Name | Procedure/Function | Description |
| TBD | TBD | TBD |

6.2.2.3.22. Special References

Special Reference will be developed as development proceeds.

Table 55 Sipec•.aI References

|  |  |  |
| --- | --- | --- |
| Special Reference Name | Type | Description |
| TBD | TBD | TBD |

6.2.2.3.23. Class Events

Class Event will be developed as development proceeds.

Table 56: Class Events

I I

I Name Type Description

TBD

TBD TBD

6.2.2.3.24. Class Methods

TBD as development proceeds

Table 57: Class Methods

|  |  |  |
| --- | --- | --- |
| Name | Procedure/Function | Description |
| TBD | TBD | TBD |

6.2.2.3.25. Class Properties

TBD as development proceeds

Table 58: Class Properties

|  |  |  |  |
| --- | --- | --- | --- |
| Class Properties Name | Type | Visibility | Description |
| TBD | TBD | TBD | TBD |

6.2.2.3.26. Uses Clause

Details will be defined during the development cycle and are not applicable at this time.

6.2.2.3.27. Forms

Details will be defined during the development cycle and are not applicable at this time.

Table 59: Forms (Instructions)

|  |  |
| --- | --- |
| Forms | Instructions |
| Form Name | TBD |
| Enhancement Category | TBD |
| Form Functionality | TBD |
| Current Form Layout | TBD |
| Modified Form Layout  (Changes are in bold) | TBD |

Table 60 (Grouping): Forms

|  |  |
| --- | --- |
| Forms | Description |
| Form Name | TBD |
| Enhancement Category | D New D Modify D Delete D No Change |
| Form Functionality | TBD |

I Current Form Layout

TBD

Modified Form Layout (Changes are in bold)

TBD

6.2.2.3.28. Functions

Details will be defined during the development cycle and are not applicable at this time.

Table 61: Functions (Instructions)

|  |  |
| --- | --- |
| Functions | Instructions |
| Function Name | TBD |
| Short Description | TBD |
| Enhancement Category | TBD |
| Related Options | TBD |
| Related Routines | TBD |
| Data Dictionary (DO) References | TBD |

|  |  |
| --- | --- |
| Functions | Instructions |
| Related Protocols | TBD |
| Related Integration Control  Registrations (ICRs) | TBD |
| Data Passing | TBD |
| Input Attribute Name and  Definition | TBD |
| Output Attribute Name and  Definition | TBD |
| Current Logic | TBD |
| Modified Logic (Changes are in bold) | TBD |

Table 62 (Grouping): Functions

|  |  |
| --- | --- |
| Function Name | Activities |
| Short  Description | TBD |
| Enhancement  Category | DNew D Modify D Delete D No Change |
| Related Options | TBD |

|  |  |  |
| --- | --- | --- |
| Related Routines | Routines "Called By" | Routines "Called" |
| **TBD** | TBD | TBD |

|  |  |
| --- | --- |
| Function Name | Activities |
| Data Dictionary  **(DO)** References | TBD |
| Related  Protocols | TBD |
| Related Integration Control Registrations (ICRs) | TBD |
| Data Passing | D Input D D Both D Global D Local Reference  Output Reference |
| Input Attribute | Name: |

|  |  |
| --- | --- |
| Function Name | Activities |
| Name and  Definition | Definition: |
| Output Attribute Name and Definition | Name: Definition: |

I Current Logic

TBD

Modified Logic (Changes are in bold)

TBD

6.2.2.3.29. Dialog

Dialogs are described in the "To-Be" requirements and will be developed as development proceeds.

Table 63: Dialog (Instructions)

|  |  |
| --- | --- |
| Dialog | Instructions |
| Dialog Message  (Description) | TBD |
| Enhancement Category | TBD |
| Dialog Message  (Description) Condition | TBD |
| Current Dialog Message  (Description) | TBD |
| Modified Dialog Message  (Description) (Changes are in bold) | TBD |

Table 64 (Grouping): Dialog

|  |  |
| --- | --- |
| Dialog | Instructions |
| Dialog Message  (Description) | TBD |
| Enhancement Category | D Modify D Delete D No Change |
| Dialog Message  (Description) Condition | TBD |
| Current Dialog Message  (Description) | TBD |

|  |  |
| --- | --- |
| Dialog | Instructions |
| Modified Dialog Message  (Description) (Changes are in bold) | TBD |

6.2.2.3.30. Help Frame

Details will be defined during the development cycle and are not applicable at this time.

Table 65: Help Frame (Instructions)

|  |  |
| --- | --- |
| Help Frame | Instructions |
| Help Frame Text | TBD |
| Enhancement Category | TBD |
| Help Frame Text Calling  Mechanism | TBD |
| Current Help Frame Text | TBD |
| Modified Help Frame Text  (Changes are in bold) | TBD |

Table 66 (Grouping): Help Frame

|  |  |
| --- | --- |
| Help Frame | Description |
| Help Frame Text | TBD |
| Enhancement Category | D Modify D Delete D No Change |
| Help Frame Text Calling  Mechanism | TBD |

I Current Help Frame Text

TBD

Modified Help Frame Text (Changes are in bold)

TBD

6.2.2.3.31. HL7 Application Parameter

Details will be defined during the development cycle and are not applicable at this time.

Table 67: HL7 Application Parameter (Instructions)

|  |  |
| --- | --- |
| HL7 Application Parameter | Instructions |
| HL7 Application Parameter  Name | TBD |
| Enhancement Category | TBD |

|  |  |
| --- | --- |
| HL7 Application Parameter | Instructions |
| Application Status | TBD |
| Facility Name | TBD |
| Country Code | TBD |
| HL7 Field Separator | TBD |
| HL7 Encoding Characters | TBD |
| Mail Group | TBD |

HL7 Application

Parameter Name

I Enhancement Category

Table 68: HL7 Application Parameter

Description

I D New D Modify D Delete D No Change

I Application Status I D Active Dlnactive D Active Dlnactive

|  |  |  |
| --- | --- | --- |
| Enhancement Category | Current | Modified |
| Facility Name | TBD | TBD |
| Country Code | TBD | TBD |
| HL7 Field Separator | TBD | TBD |
| HL7 Encoding Characters | TBD | TBD |
| Mail Group | TBD | TBD |

6.2.2.3.32. HL7 Logical Link

Table 69: HL7 Logical Link (Instructions)

|  |  |
| --- | --- |
| HL7 Logical Link | Instructions |
| HL7 Logical Link Parameter  (LLP) Name | TBD |
| Enhancement Category | TBD |
| Node | TBD |
| Institution | TBD |
| Domain | TBD |
| Autostart | TBD |
| Queue Size | TBD |
| LLP Type | TBD |

Table 70 HL7 LOQICaI L"Ink

|  |  |  |
| --- | --- | --- |
| HL7 Logical Link | Description | |
| HL7 Logical Link  Parameter Name | TBD | |
| I Enhancement Category | I D New D Modify D Delete D No Change | |
| Enhancement Category | Current | Modified |
| Node | TBD | TBD |
| Institution | TBD | TBD |
| Domain | TBD | TBD |
| Autostart | TBD | TBD |
| Queue Size | TBD | TBD |
| LLP Type | TBD | TBD |

6.2.2.3.33. COTS Interface

Table 71: COTS Interface (Instructions)

|  |  |
| --- | --- |
| COTS Interface | Instructions |
| Communication Method | TBD |
| Application Interface | TBD |

Table 72·COTS Interface

|  |  |
| --- | --- |
| COTS Interface | Description |
| Communication Method | TBD |
| Application Interface | TBD |

6.3. Network Detailed Design

The CIT&DM uses standard network devices such as routers, switches, fuewalls, and load balancers implanted at AITC. Detailed specification on these network devices are not provided here. The discussion herein focuses on two types of content transfer: data backup and deployment from development environment to user acceptance testing environment.

There are two major types of content transfers between servers: backup ente1prise data from tetminology development server to backup database server and data from development environment to pre-production or test environment. Policies of both data backup policy and content transfer are needed to minimize the dismption of operations.

**6.4. Security and Privacy**

**6.4.1. Security**

The CTT&DM will adhere to all VA security requirements in accordance to VA Directive and Handbook 6500, Federal Information Processing Standard (FIPS) 199 and National Institute of Standards and Technology (NIST) SP 800-60, recommended Security Categorization. CTT&DM will be housed at the Austin Information Technology Center (AITC), and information security controls are covered under the AITC General Support System (GSS). A continuous monitoring plan has been implemented which reviews key security controls for regular and continuous monitoring.

The CTT&DM Security Categorization will drive the initial set of moderate security controls required for the information system. Minimum security control requirements are addressed in NIST SP 800-53 Revision 4 and VA Handbook 6500, March 2015, Appendix C: (References), Appendix E: (VA System Privacy Controls), Appendix F: (VA System Security Controls).

**6.4.2. Privacy**

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

The CTT&DM 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) and/or Protected Health Information (PHI) must be covered by a Privacy Act system of records notice.

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

CTT&DM uses standard network devices such as routers, switches, firewalls, and load balancers implanted at AITC. Detailed specifications for these network devices are not provided here. The discussion herein focuses on two types of content transfer:

1. Data backup.

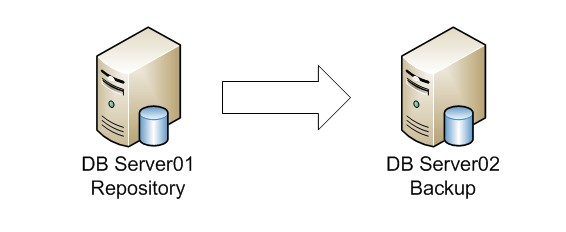
2. Deployment from development environment to user acceptance testing environment. There are two major types of content transfer between servers:

1. Backup enterprise data from terminology development server to backup database server.

2. Data from development environment to preproduction or test environment.

Policies governing both data backup and content transfer are needed to minimize the disruption of operations.

The Data Backup Flow Diagram is depicted in the figure below and shows the basic data flow during typical data backup operations.



**Figure 22: Data Backup Flow Diagram**

The growth of data volumes necessitates an efficient approach to data backup and recovery. For any organization that is highly dependent on Information Technology (IT), backup and maintenance of data for servers, are critical factors for its viability and operations. It is essential that certain basic standard practices be followed to ensure that data files are backed up on a regular basis.

The Systems Support team ensures that all backups are completed successfully and reviews the backup process on all servers daily. Logs are maintained to verify the amount of data backed up and the unsuccessful backup occurrences. Once the new developed system is in place, backup procedures will be relatively the same.

**Backup Content**

The content of data backed up varies from server to server. The primary data that will be backed up are XML files designated to be backed up in the background as a task to be done on a regular basis. Data to be backed up will be listed by location and specified data sources. This will be stipulated in a separate document called the “Data Sources Manifest”. Because it is impractical for the Systems Support to backup every bit of data stored on the servers, the only data that

Systems Support accepts responsibility for is the data which is explicitly listed in the “Data

Source Manifest”.

**Backup Types**

Backup of servers may occur every day after regular business hours or backups may occur per other regularly scheduled backup plan. Three types of backup are listed below:

x **Full backup.** Includes all the source files. This method ignores the file's archive bit until after the file is backed up. At the end of the job, all files that have been backed up have their archive bits turned off. Only one full backup will be done per week, followed by differential and/or incremental.

x **Differential backup.** Includes files that have been changed since the last Full (Clear Archive Bit) or Incremental backup. If the archive bit is on, the file is backed up and the archive bit is not turned off. The next time an incremental backup is done, this file is skipped (unless it has been modified again).

x **Incremental backup.** Includes only files that have changed since the last Full (Clear Archive Bit) or Incremental backup. The next time an incremental backup is done, this file is skipped (unless it has been modified again).

Daily backups take place on a five-day rotation. Weekly backups take place on a five-week rotation. Monthly backups of high availability servers occur the last calendar day of the month and are on a twelve-month rotation. Special backups may be made for longer retention periods during special situations, such as system upgrades and major projects.

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

Details will be defined during the development cycle and are not applicable at this time.

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

Details will be defined during the development cycle and are not applicable at this time.

**6.5.2.1. Introduction**

**6.5.2.1.1. Purpose and Scope of Service**

Details will be defined during the development cycle and are not applicable at this time.

**6.5.2.1.2. Links to Other Documents**

Details will be defined during the development cycle and are not applicable at this time.

**6.5.2.2. Service Details**

**6.5.2.2.1. Service Identification**

Details will be defined during the development cycle and are not applicable at this time.

Table 73: Service Attributes

|  |  |
| --- | --- |
| Service Attribute | Value |
| Name and Alias (if any) | TBD |
| Overview | TBD |
| Version | TBD |
| Latest Status | TBD |
| Service Type | TBD |
| Architecture Layer | TBD |
| Business Domain | TBD |
| Service Domain | TBD |
| Business Organization and Owner | TBD |
| Technical Organization and Owner | TBD |
| Development Organization and  Owner | TBD |
| Support Organization and Owner | TBD |
| Target Consumer Organization(s)  and Owner(s) | TBD |

6.5.2.2.2. Service Versions

Table 74: Service Versions

|  |  |  |
| --- | --- | --- |
| Version Numbers | Current Status of  Version | A Brief Description of the change implemented in that version |
| TBD | TBD | TBD |
| TBD | TBD | TBD |
| TBD | TBD | TBD |

6.5.2.2.3. Summary of Design and Platform Details

6.5.2.2.3.1. SOA Pattern(s) Implemented

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.2.3.2. COTS Platform Vendor Names and Versions for Hosting Platform

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.3. Dependencies

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4. Service Design Details

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.1. Interface Technical Specs

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.1.1. Service Invocation Type

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.1.2. Service Interface Type

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.1.3. Service Name

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.1.4. Interface

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.1.5. End Points

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.1.6. Operations or Methods

Details will be defined during the development cycle and are not applicable at this time.

Table 75 01perar1on or Meth0ds

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

6.5.2.4.1.7. Message Schemas

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.2. Information Model

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.2.1. Class Diagram and Description of Entities Involved

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.2.2. Mappings from ELDM to Standards Based Schemas

Details will be defined during the development cycle and are not applicable at this time.

6.5.2.4.3. Behavior Model (AKA Use Case Realization)

Details will be defined during the development cycle and are not applicable at this time.

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

**Concept Workflow**

**A. Create a Request for a Change to Concept(s)**

**User Type: Editor**

1. The need to create a new concept or a change an existing concept

2. An editor creates a request for the new concept to be created or for the existing concept to be modified.

3. The request is submitted for review.

4. The request is routed to the appropriate reviewer Queue.

5. The request is marked as “Assigned”.

6. The request resides in the assigned reviewer Queue.

**B. Review and Reject/Approve the Concept(s)**

**User Type: Reviewer**

1. The workflow state associated with this request is marked as “Ready for Review” in the reviewer workflow inbox.

2. The request resides in the appropriate Reviewer’s Queue.

3. The Reviewer reviews the request that is marked “Ready for Review”.

4. The reviewer rejects the modification to one or more concepts made by the editor a. The reviewer marks the rejected concepts as “Rejected” with comment.

b. The reviewer sends back the request to the original editor.

5. The Concept Reviewer approves the modification

a. The reviewer marks the approved concepts as “Approved”. b. The reviewer sends the request to the approval.

c. The approved request is marked “Ready for Approve” in the assigned approval workflow inbox queue.

**C. Approve the Request**

**User Type: Approval**

1. The workflow state associated with this request is marked as “Ready to Approve”

2. The request resides in the approver’s Queue.

3. The approver reviews the request and the concept(s) marked “Ready to Approve”.

4. The approver approves the modification to all concepts made by the original editor.

5. The approver marks all concepts as “Approved”.

6. Requests are marked as “Completed”.

7. All concepts are marked as “Approved”.

8. The requests are removed from the approver’s Queue.

9. The approved changes are published and system updated accordingly.

**D. Reject the Request**

**User Type: Approver**

1. The workflow state associated with this request is marked as “Ready to Approve”.

2. The request resides in the approver’s Queue.

3. The approver reviews the request and the concept(s) marked “Ready to Approve”.

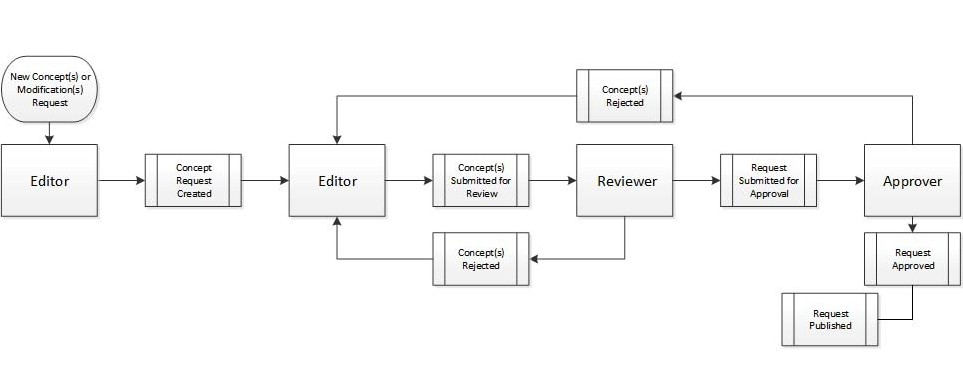
4. The approver rejects the modification to one or more concepts.

5. The approver marks the rejected concepts as “Rejected”.

6. The approver adds comments on why the request was rejected.

7. The approver sends back the request to the editor.

8. The request is removed from the approver’s Queue.



**Figure 23: New/Modified Concept Process Flow**

**Terminology Workflow**

**E. Create a Request for a Change to Terminology(s) User Type: Editor**

1. The need to create a new terminology or to change an existing terminology

2. An editor creates a request for the new terminology to be created or for the existing terminology to be modified.

3. The request is submitted for review.

4. The request is routed to the appropriate reviewer Queue.

5. The request is marked as “Assigned”.

6.The request resides in the assigned reviewer Queue.

**F. Review and Reject/Approve the Terminology(s)**

**User Type: Reviewer**

1. The workflow state associated with this request is marked as “Ready for Review” in the reviewer workflow inbox.

2. The request resides in the appropriate Reviewer’s Queue.

3. The Reviewer reviews the request that is marked “Ready for Review”.

4. The reviewer rejects the modification to one or more terminologies made by the editor

5. The reviewer marks the rejected terminology as “Rejected” with comment.

6. The reviewer sends back the request to the original editor.

7. The Terminology Reviewer approves the modification

8. The reviewer marks the approved terminology as “Approved”.

9. The reviewer sends the request to the approval.

10. The approved request is marked “Ready for Approve” in the assigned approval workflow inbox queue.

**G. Approve the Request**

**User Type: Approval**

1. The workflow state associated with this request is marked as “Ready to Approve”

2. The request resides in the approver’s Queue.

3. The approver reviews the request and the terminology(s) marked “Ready to

Approve”.

4. The approver approves the modification to all terminology made by the original editor.

5. The approver marks all terminology as “Approved”.

6. Requests are marked as “Completed”.

7. All terminology is marked as “Approved”.

8. The requests are removed from the approver’s Queue.

9. The approved changes are published and system updated accordingly.

**H. Reject the Request**

**User Type: Approval**

1. The workflow state associated with this request is marked as “Ready to Approve”.

2. The request resides in the approver’s Queue.

3. The approver reviews the request and the terminology(s) marked “Ready to

Approve”.

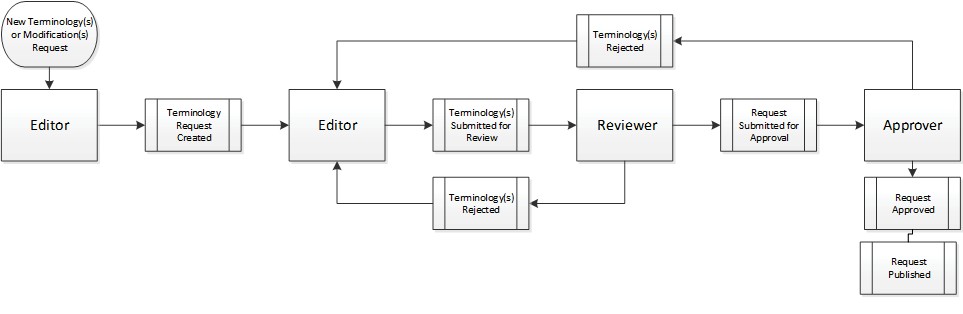
4. The approver rejects the modification to one or more terminology.

5. The approver marks the rejected terminology as “Rejected”.

6. The approver adds comments on why the request was rejected.

7. The approver sends back the request to the editor.

8. The request is removed from the approver’s Queue.



**Figure 24: Terminology Workflow**

**Read-Only Workflow**

**“To-Be”**

**I. Create a Request for a Change to Terminology(s)**

**User Type: Read-Only User**

1. The need to create a new terminology or a change to an existing terminology

2. A read-only user creates a request for the new terminology to be created or for the existing terminology to be modified (this is done in a tracking system i.e., Jira)

3. The request is submitted for review.

4. The request is routed to the appropriate reviewer Queue.

5. The request is marked as “Assigned”.

6. The request resides in the assigned reviewer Queue.

**J. Review and Reject/Approve the Terminology(s)**

**User Type: Reviewer**

1. The workflow state associated with this request is marked as “Ready for Review” in the reviewer workflow inbox.

2. The request resides in the appropriate Reviewer’s Queue.

3. The Reviewer reviews the request that is marked “Ready for Review”.

4. The reviewer rejects the modification to one or more terminologies made by the read only user.

5. The reviewer marks the rejected terminology as “Rejected” with comment.

6. The rejected request is sent back to the read only user (via tracking system)

7. The read only user receives an email notification of the status update (via tracking system)

8. The Terminology Reviewer approves the modification

9. The reviewer marks the approved terminology as “Approved”.

10. The reviewer sends the request to the approval.

11. The approved request is marked “Ready for Approve” in the assigned approval workflow inbox queue.

**K. Approve the Request**

**User Type: Approval**

1. The workflow state associated with this request is marked as “Ready to Approve”

2. The request resides in the approver’s Queue.

3. The approver reviews the request and the terminology(s) marked “Ready to

Approve”.

4. The approver approves the modification to all terminology made by the original editor.

5. The approver marks all terminology as “Approved”.

6. Requests are marked as “Completed”.

7. Terminology is marked as “Approved”.

8. The requests are removed from the approver’s Queue.

9. The approved changes are published and system updated accordingly.

10. The status is updated in the tracking system to notify user of the approved request.

**L. Reject the Request**

**User Type: Approval**

1. The workflow state associated with this request is marked as “Ready to Approve”.

2. The request resides in the approver’s Queue.

3. The approver reviews the request and the terminology(s) marked “Ready to

Approve”.

4. The approver rejects the modification to one or more terminology.

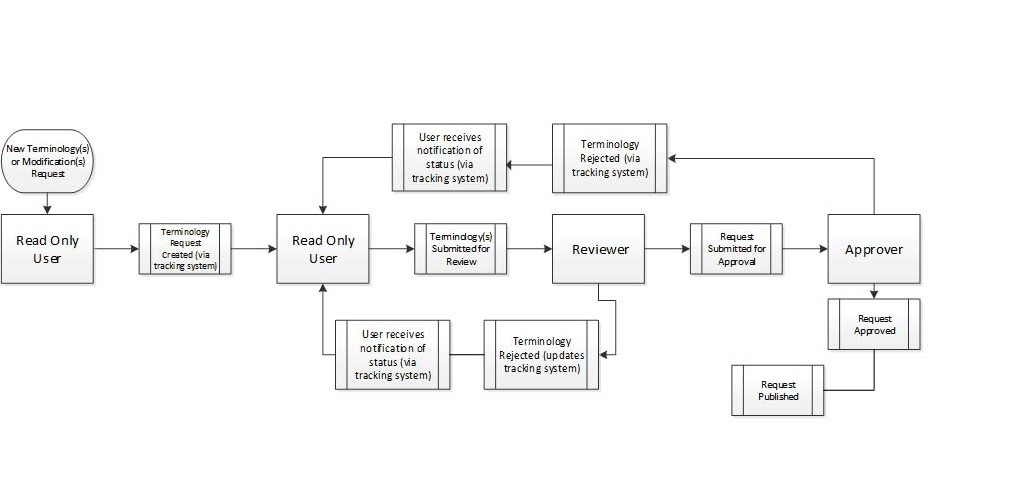
5. The approver marks the rejected terminology as “Rejected”.

6. The approver adds comments on why the request was rejected.

7. The rejected request is sent back to the read only user (via tracking system)

8. The read only user receives an email notification of the status update (via tracking system)

9. The request is removed from the approver’s Queue.



**Figure 25: “To-Be” Read-Only User Workflow**

**6.5.2.4.3.2. Interaction Diagrams**

x Step 1 of TDS. Content is received and stored in local database. Terminologists load content and manually convert them to XML files to populate them automatically for inclusion to the Repository.

x Step 2. Terminologies do internal test directly from repository. If pass the test then go to next step

x Step 3. SQA test. This test happens in VistA preproduction environment by deploying the tested content to different VistA test production systems.



**Figure 26: Continuous Integration Cycle**

A Continuous Integration (CI) server that allows you to implement the Continuous Integration

Cycle approach (see Figure 26).

Using a web-based continuous integration server allows the organization to automate the terminology build chain, e.g., monitoring changes in version control systems, triggering new builds, testing artifacts, sending notifications, deploying to production servers, etc. CI is a software development practice which allows members of the team to integrate their work frequently. Each terminology contributor may integrate daily, which means that multiple integrations may be made per day. Each integration may be verified by an automated build - including test - to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive terminology content more rapidly.

The Continuous Integration Cycle entails these basic process steps (Figure 26):

1. Developers check in new code into the version control system.

2. The CI server may poll this system regularly for changes, e.g., every minute.

3. If changes are indicated, the new code is checked out to build severs, and a new build of your project is executed.

4. The results are reported back to the CI server.

5. The CI server sets the new status of the project as stable, instable, or failed.

6. Notifications are sent to users and other stakeholders, who will either fix any problems encountered or proceed implementing new features.

Note: This complete cycle may be very fast-paced. It may not exceed much more than 15 minutes, in a typical CI server environment.

t(l!lrsi.W! I (1m

(e.g, Sub\·n)

./'

t!l:Jm act. ,

l llt.:n\_ Cllttl'IS.,

mllrlaigCintYJI

01C.

roed Cl $CI'\IIIf

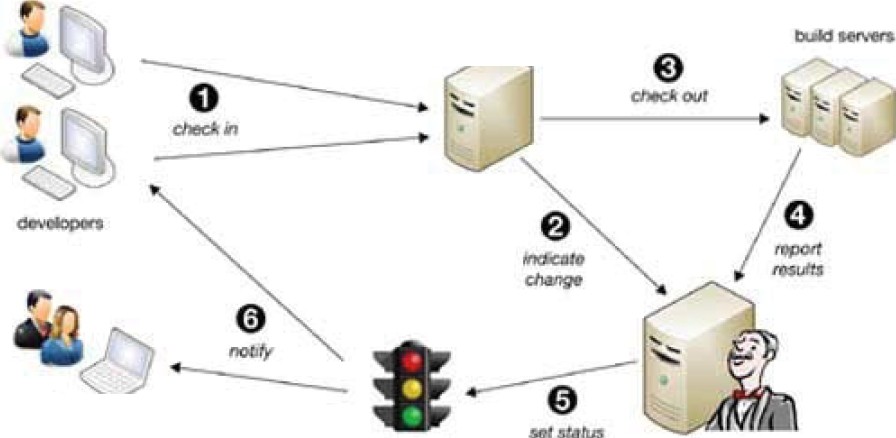


Figure 27: Continuous Integration Cycle Steps

6.5.2.5. Gap Analysis

Table 76 Gap AnaiYSIS

|  |  |  |  |
| --- | --- | --- | --- |
| Category | Baseline | Gap Description | Artifact |
| Accessibility  Specifications | No Changes | No Changes | N/A |
| Design Constraints  Specifications | No Changes | No Changes | N/A |
| Disaster Recovery  Specification | No Disaster  Documentation | Disaster specifications have been added. | RSD/ Section 2.4 |
| Documentation  Specifications | N/A | Document specifications have been added. | RSD/ Section 2.5 |
| System Functions | N/A | The system should be able to check for duplication upon conception. | RSD/Section 2.6.3.1 |
| System Functions | N/A | The system should run a daily QA process checking for duplicates. | RSD/Section 2.6.3.1 |
| System Functions | N/A | The system should support the functionality require to convert terms to common required languages. | RSD/Section 2.6.3.1 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| System Functions | N/A | The system should allow a user to be assigned multiple roles | RSD/Section 2.6.3.1 |
| System Functions | N/A | The system shall allow users to be assigned concepts | RSD/Section 2.6.3.1 |
| System Functions | N/A | The system shall allow the workflow status to be edited | RSD/Section 2.6.3.1 |
| System Functions | N/A | The system shall allow additional workflow status to be added (in addition to the prototype list) | RSD/Section 2.6.3.1 |
| System Functions | N/A | The system should support the functionality require to convert terms to common required language. | RSD/Section 2.6.3.1 |
| Alerts | No Changes | No Changes | N/A |
| Identity and Access  Management Service | No Changes | No Changes | N/A |
| Human Consults | No Changes | No Changes | N/A |
| Context-sensitive help | No Changes | No Changes | N/A |
| Workflow | No Changes | No Changes | N/A |
| Terminology Releases from Standards Development Organizations (SDO) | No Changes | No Changes | N/A |
| Terminology Releases from Knowledge Sharing Partners | No Changes | No Changes | N/A |
| Terminology Releases from Knowledge Sharing Partners | No Changes | No Changes | N/A |
| XML Transforms | No Changes | No Changes | N/A |
| Outputs From System | No Changes | No Changes | N/A |
| Publishing Output  Content (VETS) | No Changes | No Changes | N/A |
| Business-Specified Non-Functional Requirements | No Changes | No Changes | N/A |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Document Generation  Capabilities | No Changes | No Changes | N/A |
| Mapping Capabilities | N/A | The tooling shall allow authors to store guidelines or rules (heuristics) that govern the map and its creation and use. | RSD/ Section 1.2  Increment 1  Requirement |
| Mapping Capabilities | N/A | The tooling shall allow the capability to update the map set when the source and target are updated from standards development organizations. | RSD/ Section 1.2  Increment 1  Requirement |
| Mapping Capabilities | N/A | The tooling shall allow authors to define the metadata for a map seUvalue set. | RSD/ Section 1.2  Increment 1  Requirement |
| Browse a Set of Approved, Suggested or Similar Models | No Changes | No Changes | N/A |
| Template Based  Modeling | No Changes | No Changes | N/A |
| Modify Existing | No Changes | No Changes | N/A |
| Specialize From  Existing | No Changes | No Changes | N/A |
| Duplicate Pieces and Re-use in a New Model | No Changes | No Changes | N/A |
| Rules Driven  Modeling | No Changes | No Changes | N/A |
| Interview Driven  Modeling | No Changes | No Changes | N/A |
| Rules Driven QA | No Changes | No Changes | N/A |
| Semi-Automated  Modeling | No Changes | No Changes | N/A |
| Stratified by Domain | No Changes | No Changes | N/A |
| Stratified by  Architectural Level | No Changes | No Changes | N/A |
| Workflow  Management Service | No Changes | No Changes | N/A |
| Taxonomy View | No Changes | No Changes | N/A |
| Description Logic | No Changes | No Changes | N/A |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Quality Assurance | No Changes | No Changes | N/A |
| Additional  Considerations | No Changes | No Changes | N/A |
| User Type | No User Role  Documentation | Read-Only User | RSD/ Section  2.6.3.26 User Type |
| User Type | No User Role  Documentation | Editor User | RSD/ Section  2.6.3.26 User Type |
| User Type | No User Role  Documentation | Reviewer User | RSD/ Section  2.6.3.26 User Type |
| User Type | No User Role  Documentation | Approver | RSD/ Section  2.6.3.26 User Type |
| User Type | No User Role  Documentation | Final Approval User | RSD/ Section  2.6.3.26 User Type |
| User Type | No User Role  Documentation | Administrative User | RSD/ Section  2.6.3.26 User Type |
| User Functionality | N/A | User shall be assigned to a concept and a role | RSD/ Section 2.6.3.7  User Functionality |
| Task Request | N/A | User shall be able to reject or approve task/request | RSD/ Section  2.6.3.28 Task  Request |
| Task Request | N/A | User shall receive email notifications notifying the user a task/request is available | RSD/ Section  2.6.3.28 Task  Request |
| Task Request | N/A | User shall be able to retrieve task/request | RSD/ Section  2.6.3.28 Task  Request |
| Task Request | N/A | User shall be able to reject or approve task/request | RSD/ Section  2.6.3.28 Task  Request |
| Task Request | N/A | User shall be able to filter task/request by open, closed, due date, etc. | RSD/ Section  2.6.3.28 Task  Request |
| Task Request | N/A | User shall be able to reject or approve task/request | RSD/ Section 2.6.33  Task Request |
| Task Request | N/A | User shall receive email notifications notifying the user a task/request is available | RSD/ Section 2.6.33  Task Request |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Terminology | N/A | User shall have the ability to add new terms, relationships and attributes to represented terminologies | RSD/ Section  2.6.3.29 Terminology |
| Terminology | N/A | User shall have the ability to retire terms, relationships and attributes from represented terminologies | RSD/ Section  2.6.3.29 Terminology |
| Terminology | N/A | User shall have the ability to modify terms, relationships and attributes in represented terminologies | RSD/ Section  2.6.3.29 Terminology |
| Terminology | N/A | User should have the ability to browse previous versions of terminologies | RSD/ Section  2.6.3.29 Terminology |
| Terminology | N/A | User shall be able to reject or approve task | RSD/ Section  2.6.3.29 Terminology |
| Terminology | N/A | User shall have the ability to compare the differences between different versions of terminologies | RSD/ Section  2.6.3.29 Terminology |
| Terminology | N/A | User shall be able to clone terminology from another terminology | RSD/ Section  2.6.3.29 Terminology |
| Concept | N/A | User shall have the ability to clone concept - The ability to create a new concept as a clone of another concept | RSD/ Section 2.6.35  Concept |
| Concept | N/A | User shall have the ability to create child concept. The ability to create a new concept as a child of another | RSD/ Section  2.6.3.30 Concept |
| Concept | N/A | User shall have the ability to move a set of concepts to a new parent concept | RSD/ Section  2.6.3.30 Concept |
| Concept | N/A | User shall have the ability to retire a concept | RSD/ Section  2.6.3.30 Concept |
| Concept | N/A | User shall have the ability to add concepts | RSD/ Section  2.6.3.30 Concept |
| Concept | N/A | User shall have the ability to modify concepts | RSD/ Section  2.6.3.30 Concept |
| Concept | N/A | User shall have the ability to add concepts | RSD/ Section 2.6.35  Concept |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Concept | N/A | User shall have the ability to modify concepts | RSD/ Section 2.6.35  Concept |
| Concept | N/A | User shall have the ability to clone concept - The ability to create a new concept as a clone of another concept | RSD/ Section 2.6.35  Concept |
| Concept | N/A | User shall have the ability to create child concept. The ability to create a new concept as a child of another | RSD/ Section 2.6.35  Concept |
| Concept | N/A | User shall have the ability to move a set of concepts to a new parent concept | RSD/ Section 2.6.35  Concept |
| Concept | N/A | User shall have the ability to retire a concept | RSD/ Section 2.6.35  Concept |
| Concept | N/A | User shall have the ability to add concepts | RSD/ Section 2.6.35  Concept |
| Search | N/A | User shall be able to perform a basic or advance search | RSD/ Section  2.6.3.31 Search |
| Search | N/A | User shall be able to search on complete or partial name | RSD/ Section  2.6.3.31 Search |
| Search | N/A | User shall be able to search within a specified portion of the hierarchy, within reference sets, or relationship type, and destination | RSD/ Section  2.6.3.31 Search |
| Search | N/A | User shall be able to save search criteria for individual and public use | RSD/ Section  2.6.3.31 Search |
| Search | N/A | User shall be able to search for inactive and active terminology | RSD/ Section  2.6.3.31 Search |
| Search | N/A | User shall have a preference option on how the user wants to search within the taxonomy view | RSD/ Section  2.6.3.31 Search |
| Search | N/A | User shall have the ability to compare versions of text | RSD/ Section  2.6.3.31 Search |
| Search | N/A | User shall be able to perform a basic or advance search | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to search on complete or partial name | RSD/ Section 2.6.36  Search |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Search | N/A | User shall be able to search within a specified portion of the hierarchy, within reference sets, or relationship type, and destination | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to save search criteria for individual and public use | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to search for inactive and active terminology | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall have a preference option on how the user wants to search within the taxonomy view | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall have the ability to compare versions of text | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to perform a basic or advance search | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to search on complete or partial name | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to save search criteria for individual and public use | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to search for inactive and active terminology | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall have a preference option on how the user wants to search within the taxonomy view | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall have the ability to compare versions of text | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to perform a basic or advance search | RSD/ Section 2.6.36  Search |
| Search | N/A | User shall be able to search on complete or partial name | RSD/ Section 2.6.36  Search |
| Workflow lnbox | N/A | User shall have access to their personal inbox to work on task/request | RSD/ Section  2.6.3.32 Workflow  Inbox |
| Workflow lnbox | N/A | User shall be able to search and filter personal inbox | RSD/ Section  2.6.3.32 Workflow  Inbox |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Workflow lnbox | N/A | Providing the ability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. | RSD/ Section  2.6.3.32 Workflow  Inbox |
| Workflow lnbox | N/A | User shall be able to search and filter personal inbox | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | Providing the ability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall have access to their personal inbox to work on task/request | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall be able to search and filter personal inbox | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | Providing the ability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall have access to their personal inbox to work on task/request | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall be able to search and filter personal inbox | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | Providing the ability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall have access to their personal inbox to work on task/request | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall be able to search and filter personal inbox | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | Providing the ability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. | RSD/ Section 2.6.37  Workflow lnbox |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Workflow lnbox | N/A | User shall have access to their personal inbox to work on task/request | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall be able to search and filter personal inbox | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | Providing the ability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall have access to their personal inbox to work on task/request | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | User shall be able to search and filter personal inbox | RSD/ Section 2.6.37  Workflow lnbox |
| Workflow lnbox | N/A | Providing the ability to define a workflow and business process module that can be modified easily to adapt to changes in VA business requirements. | RSD/ Section 2.6.37  Workflow lnbox |
| Query | No Changes | No Changes | N/A |
| Refset Capabilities | N/A | User shall have the ability to create refsets composed of associations | RSD/ Section  2.6.3.34 Refset  Capabilities |
| Refset Capabilities | N/A | User shall have the ability to create of refsets through definitional constructs | RSD/ Section  2.6.3.34 Refset  Capabilities |
| Refset Capabilities | N/A | User shall have the ability to update existing refsets | RSD/ Section  2.6.3.34 Refset  Capabilities |
| Refset Capabilities | N/A | User shall have the ability retire terms, relationships, and attributes from refsets | RSD/ Section  2.6.3.34 Refset  Capabilities |
| Refset Capabilities | N/A | User shall have the ability to view different versions of refsets | RSD/ Section  2.6.3.34 Refset  Capabilities |
| Refset Capabilities | N/A | User shall have the ability to request and submit a request for a new refset | RSD/ Section  2.6.3.34 Refset  Capabilities |
| Refset Capabilities | N/A | User shall have the ability to request a change or modification to an existing Refset | RSD/ Section  2.6.3.34 Refset  Capabilities |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| The Project Information System and Management Environment (PRISME) | No Changes | No Changes | N/A |
| Artifact Repository  Service | No Changes | No Changes | N/A |
| Tracker Service | No Changes | No Changes | N/A |
| Service Desk | No Changes | No Changes | N/A |
| Version Control  Service | No Changes | No Changes | N/A |
| Quality Management  Platform | No Changes | No Changes | N/A |
| Project Creation  Application | No Changes | No Changes | N/A |
| Component Request  Service | No Changes | No Changes | N/A |
| Terminology Web Application Environment | No Changes | No Changes | N/A |
| New Concept  Request Application | No Changes | No Changes | N/A |
| Scripting Capabilities | No Changes | No Changes | N/A |
| General GUI Compliance | No Changes | No Changes | N/A |
| GUI Compliance Specific to Terminology Tooling | No Changes | No Changes | N/A |
| Multi-divisional  Specifications | No Changes | No Changes | N/A |
| Performance  Specifications | No Changes | No Changes | N/A |
| Quality Attributes  Specification | No Changes | No Changes | N/A |
| Reliability  Specifications | No Changes | No Changes | N/A |
| Known Interfaces | No Changes | No Changes | N/A |
| Related Projects or  Work Efforts | No Changes | No Changes | N/A |

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Baseline** | **Gap Description** | **Artifact** |
| Security  Specifications | No Changes | No Changes | N/A |
| System Features | N/A | The system should be able to check for duplication upon conception. | RSD/ Section 2.1. System Features |
| System Features | N/A | The system should run a daily QA process checking for duplicates. | RSD/ Section 2.14  System Features |
| System Features | N/A | The system should support the functionality require to convert terms to common required language. | RSD/ Section 2.14  System Features |
| System Features | N/A | The system should allow a user to be assigned multiple roles | RSD/ Section 2.14  System Features |
| System Features | N/A | The system should be able to check for duplication upon conception. | RSD/ Section 2.14.1  System Features |
| System Features | N/A | The system should run a daily QA process checking for duplicates. | RSD/ Section 2.14.1  System Features |
| System Features | N/A | The system shall allow users to be assigned concepts | RSD/ Section 2.14  System Features |
| System Features | N/A | The system shall allow the workflow status to be edited | RSD/ Section 2.14  System Features |
| System Features | N/A | The system shall allow the additional workflow status to be added (in addition to the prototype list) | RSD/ Section 2.14  System Features |
| Usability  Specifications | No Changes | No Changes | N/A |
| Applicable Standards | No Changes | No Changes | N/A |
| User Class  Characteristics | No Changes | No Changes | N/A |

**6.5.2.5.1. Variances from Enterprise Target Architecture**

Details will be defined during the development cycle and are applicable at this time.

**6.5.2.5.2. Variances from SLDs**

Details will be defined during the development cycle and are applicable at this time.

**6.5.2.5.3. Variances from Standards and Policies**

Details will be defined during the development cycle and are applicable at this time.

**6.5.2.5.4. Justification for Exceptions and Mitigation**

Details will be defined during the development cycle and are applicable at this time.

**7. External System Interface Design**

There are several key external systems that will need to be integrated into the CTT&DM design as the design progresses those interfaces will be defined in further detail here.

**7.1. Interface Architecture**

The designed system will need to initially integrate into the VA network, more importantly the existing VETS infrastructure. Details will be defined during the development cycle but the long term “To-Be” solution is to have the new system completely replace VETS.

**7.2. Interface Detailed Design**

A detailed interface design will be developed as development proceeds.

**8. Human-Machine Interface**

A detailed interface design will be developed as development proceeds.

**8.1. Interface Design Rules**

A detailed interface design will be developed as development proceeds.

**8.2. Inputs**

Standard input devices (keyboard, mouse) are needed to input data into all screens. CTT&DM is a web-based application that is not designed to be mobile.

**8.3. Outputs**

The information will be shown on the desktop or laptop monitor. Within the report screen, all reports are printable as well as can be saved in PDF, Excel, or word format.

**8.4. Navigation Hierarchy**

Details will be defined during the development cycle and are applicable at this time.

**8.4.1. Screen [x.1]**

Details will be defined during the development cycle and are applicable at this time.

**8.4.2. Screen [x.2]**

Details will be defined during the development cycle and are applicable at this time.

**8.4.3. Screen [x.3]**

Details will be defined during the development cycle and are applicable at this time.

**9. Attachment A – Approval Signatures**

This section is used to document the approval of the SDD. 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. Please sign above your name.

**A. Additional Information**

None has been identified at this time

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

See Section 4.5, Enterprise Architecture

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

CTT&DM is a standalone system but will be sharing environments with Veterans Health Information System and Technology Architecture (VistA), Consolidated Health Data Repository (CHDR), VETS, LOINC, SNOMED CT, Digital Imaging and Communication in Medicine (DICOM), DTS, Sysmedical Snow Owl, Language Engine (LE), Snapper, and Regenstreif LOINC Mapping Assistant 112 (RELMA112).

**A.3. Requirements Traceability Matrix (RTM)**

For more information please see the CTT&DM RTM.

**A.4. Packaging and Installation**

Not applicable at this time

**A.5. Design Metrics**

Not applicable at this time.