

Annie_v1.0.0_System Design Document (SDD)

Mobile Application System Design Document (SDD) Addendum

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. Per the Project Management Accountability System (PMAS) Guide, the SDD with conceptual design is required prior to the Milestone 1 Review. The as-built for each delivery must be incorporated prior to the Milestone 2 Review.

Mobile Application Information

Project Increment / Release Designation: Annie v1.0.0

Product Version: 1.0.0

Mobile Application Information

	Veteran	Care Giver	Provider	Public
Intended Audience for Mobile Application	X		X	

Question	Yes	No	If Yes, what information / data	If yes, then identify any consumer or source system(s) for the data
----------	-----	----	---------------------------------	---

Does the user enter information or data into the mobile application?	X		<p>a. Registration Data - This data can be entered by the Clinical Staff or Veteran.</p> <ul style="list-style-type: none"> • Mobile Phone • Home Phone • Email Address • Time Zone • Message Preferences (SMS or Annie Secure App) <p>b. Readings in Response to Protocol Messages - This data can be entered by the Veteran via SMS text messages or through the Annie Secure App.</p> <ul style="list-style-type: none"> • Examples: Blood Pressure, Weight, Temperature <p>c. Protocols - This data can be entered by the Clinical Admin when creating protocols or when the Clinical Staff assigns protocols.</p> <ul style="list-style-type: none"> • Schedule Settings • Readings/Alerts Settings • Message Settings <p>d. Broadcast Messages - This data can be entered by the Broadcast Admin from the Main BCM Page.</p> <ul style="list-style-type: none"> • Schedule Settings • Broadcast Message Information 	Data Stored in VAMF MongoDB instance
--	---	--	--	--------------------------------------

Does Mobile Application store information or data entered by the User? If yes, where is it stored?	X		<p>a. Registration Data</p> <ul style="list-style-type: none"> • Mobile Phone • Home Phone • Email Address • Time Zone • Message Preferences (SMS or Annie Secure App) <p>b. Readings in Response to Protocol Messages</p> <ul style="list-style-type: none"> • Examples: Blood Pressure, Weight, Temperature <p>c. Protocols</p> <ul style="list-style-type: none"> • Schedule Settings • Readings/Alerts Settings • Message Settings <p>d. Broadcast Messages</p> <ul style="list-style-type: none"> • Schedule Settings • Broadcast Message Information 	Data Stored in VAMF MongoDB instance
Does Mobile Application transmit/push data entered outside of the VAMF to VA?		X		
Does Mobile Application pull data from a VA Database (external to VAMF)?	X		<p>a. Veteran Demographics (FName, LName, SSN, Gender, DOB)</p> <p>b. Appointment Data (Date, Time, Facility, Clinic Phone Number)</p>	<p>a. Health Adapter/Patient Viewer Services - source VistA?</p> <p>b. Health Data Repository (HDR)Clinical Data Services (CDS)</p>
Does the Mobile Application store in the VAMF or on the device data pulled from a VA Database?	X		<p>a. Veteran Demographics (FName, LName, SSN, Gender, DOB)</p> <p>b. Appointment Data (Date, Time, Facility, Clinic Phone Number)</p>	Data Stored in VAMF MongoDB instance

This application can be classified as one of the following:

Mobile Application Classification (Only one box may be checked)	Mark with X
1 – Very Low: Mobile Application does not use VA Resource	
2 – Low: Read only access to VA Resource(s) (No PII / PHI)	
3 – Medium: Write access to VA Resource(s)	
4 – High: Read and/or Write access of sensitive data to VA Resource(s) (Includes PII/PHI/other sensitive)	X

Application Design

Note:

- Integration with PGD (and eCRUD) is in the backlog and is not part of this release
 - Risk Based Decision Memorandum approved
-

Network Architecture

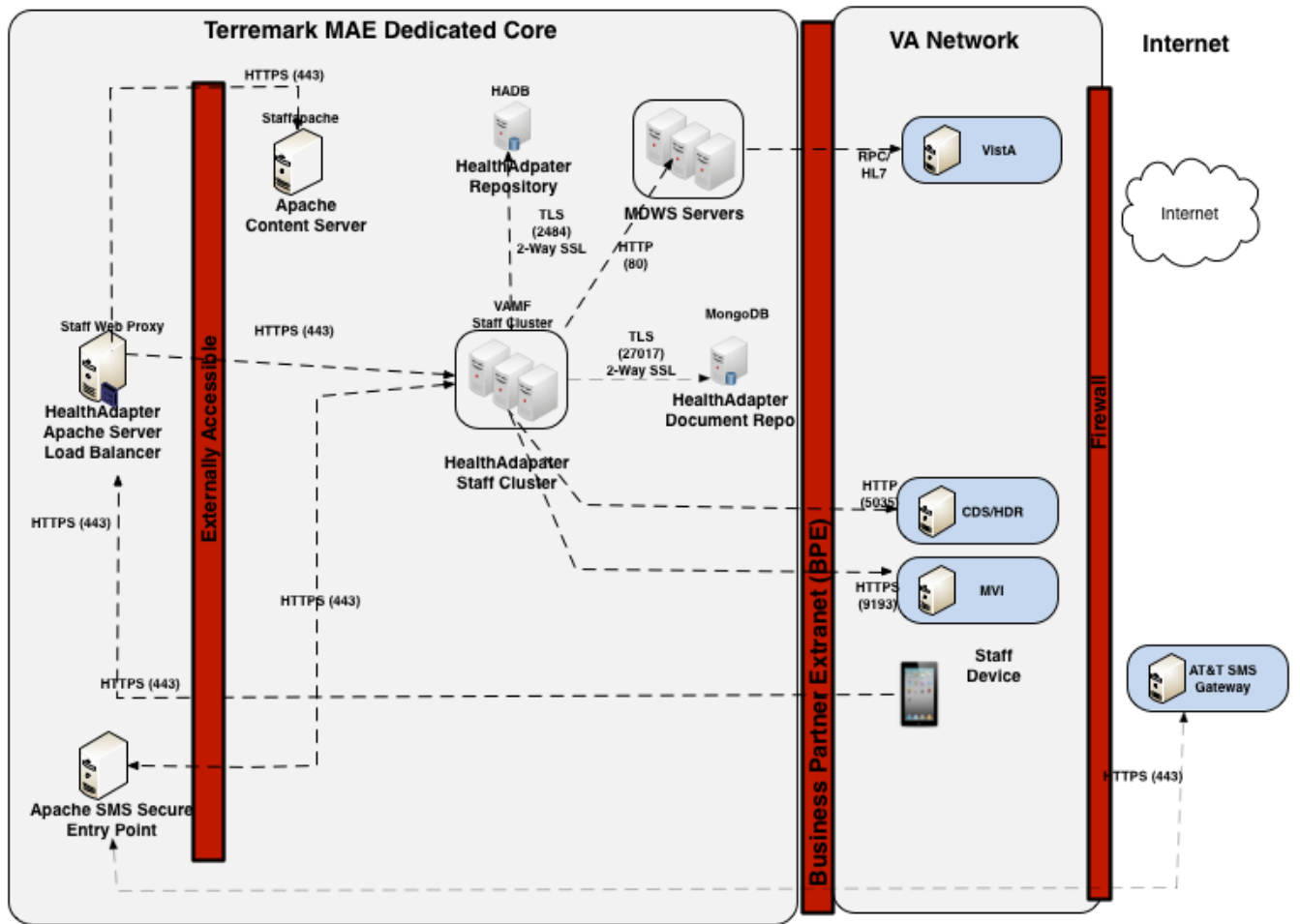
Network Overview

The CMS deployment environment is comprised of several components, including:

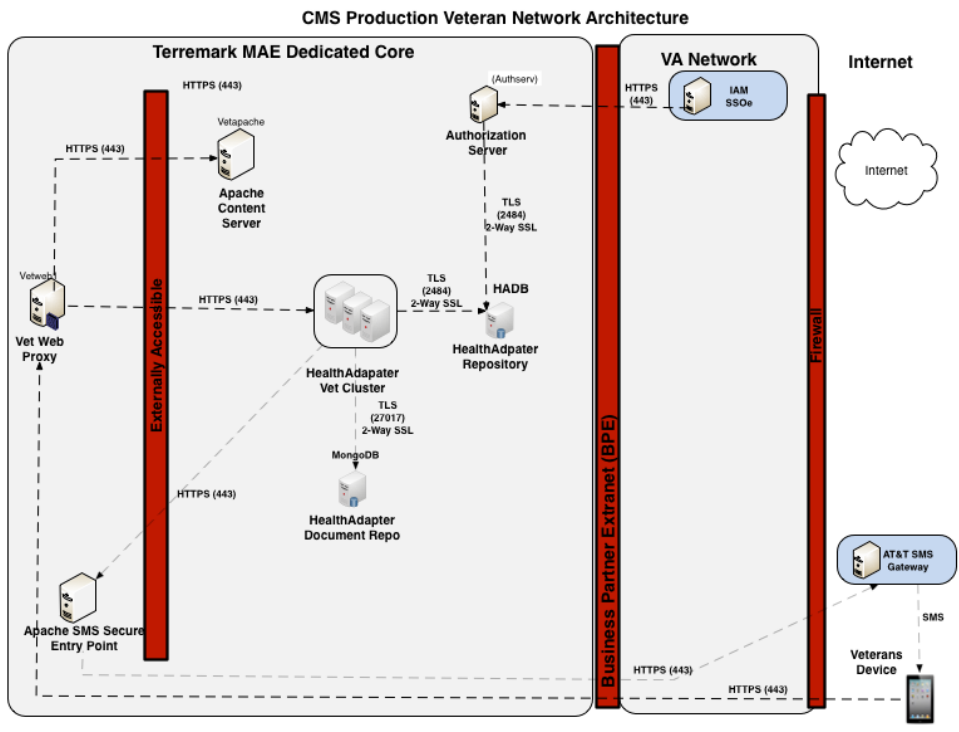
- Health Adapter Apache Server Load Balancer - A load balancer to distribute transactions among the CMS (and HealthAdapter) Nodes
- Apache Content Server - An Apache instance to serve static content files (.html, .js, .png, etc) which reduces the workload on the Application Servers
- CMS Staff and Vet Clusters - Application servers to host the CMS services configured as an application server cluster.
- HealthAdapter Staff and Vet Clusters - Application servers to host the HealthAdapter services configured as an application server cluster.
- HealthAdapter Document Repository - This is a MongoDB instance where application data is stored. This may eventually be replaced in part or in whole by the PGD when available.
- HealthAdapter Repository - A shared repository. This is an Oracle instance used to store infrastructure information such as scheduled jobs and JMS Queue Messages
- Apache SMS Secure Entry Point - An Apache instance that is used to communicate with external SMS Messaging Providers (AT&T).
- [Authorization Server](#) - Veteran System Instance only
- MVI - Provides Patient Correlation Services to determine correct veteran identifier to query CDS/HDR
- CDS/HDR - Provides Future Appointment data for sending appointment reminders
- Staff Devices - Desktop and mobile devices accessing the CMS Staff Application via a browser.
- Veteran Devices - Veteran mobile devices will communicate via SMS messages which are delivered to CMS via the AT&T SMS Gateway. Smart Phones and Desktop devices will be able to utilize the Veteran App via a browser.

Annie focused view of Staff Environment

CMS Production Staff Network Architecture



Annie focused view of Veteran environment



System Components

The following systems/sub-systems describe interfaces utilized by the project

Health Adapter Authorization Services: Services to provide Authentication and Authorization services.

[AuthServices SDD](#)

Staff Authentication: [REDACTED]

Veteran Authentication: [REDACTED]

How to use Authorization Services: [REDACTED]

Health Adapter Patient Viewer Services: Provides REST endpoint to query for patients.

The following endpoints are utilized:

PatientViewerServices/rest/patients?searchstring=

PatientViewerServices/rest/patient/{assigningAuthority}/{uniqueId}/

Details about Patient Viewer Services can be found at [REDACTED]

Oracle: Relational data storage. Used to store Quartz infrastructure data

Mongo: Document data storage. Data storage for application data

Quartz: Internal component that provides scheduling capability for "jobs". Scheduled jobs are used for scheduled text messages and for scheduling appointment reminders refreshes.

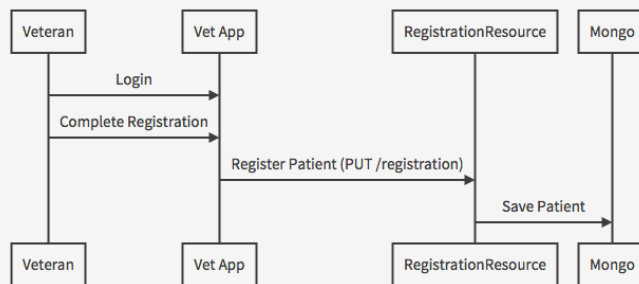
JMS (Weblogic Provider): Queuing infrastructure. Used to offset server load for text messages being received and sent. Also used for running appointment reminder refreshes.

AT&T Messaging Suite: Provides http endpoint that allows text messages to be sent to phones. It also notifies the system when SMS messages are sent from phones.

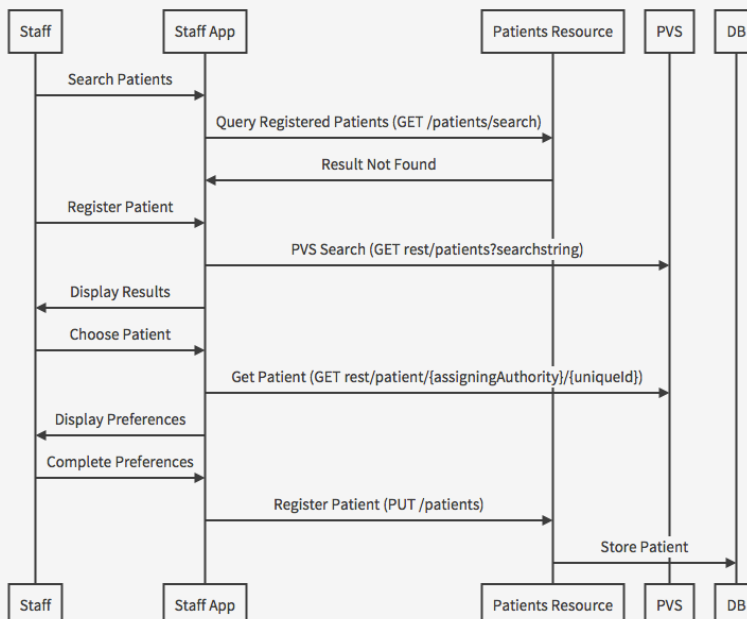
[NDA] AT&T GSMS- HTTP & WSDL Interface Developers Guide - AT&T v7.2.1.pdf

Sequence Diagrams

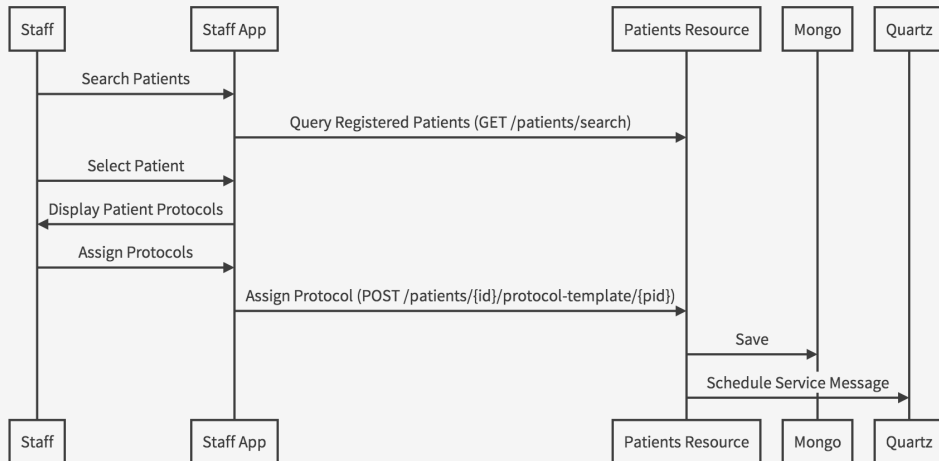
Veteran Registration - Veteran registers via the Veteran Application for Appointment Reminders and/or Broadcast Messages



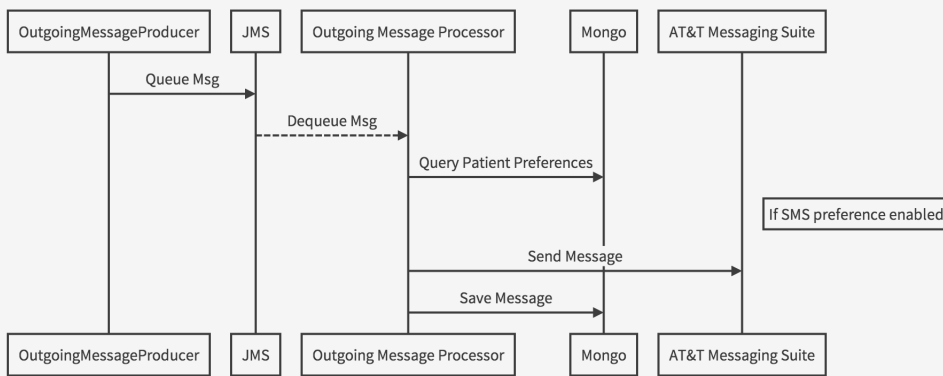
Staff Registration - Veteran is registered via Staff assisted registration in the Staff app for Protocols and/or Appointment Reminders and Broadcast Messages



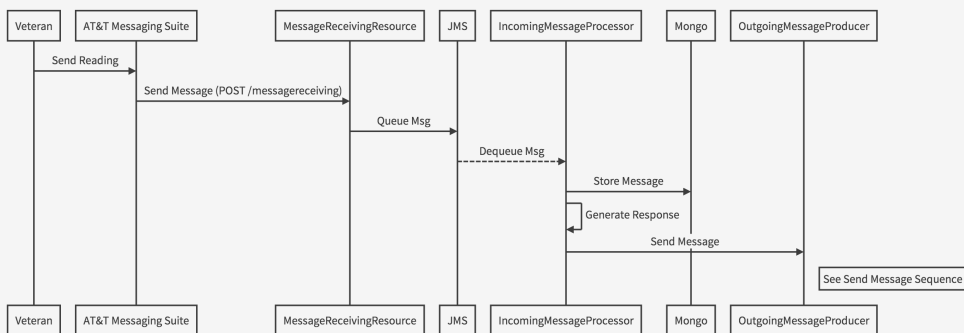
Assign Protocols - Staff assigns a patient to an existing protocol



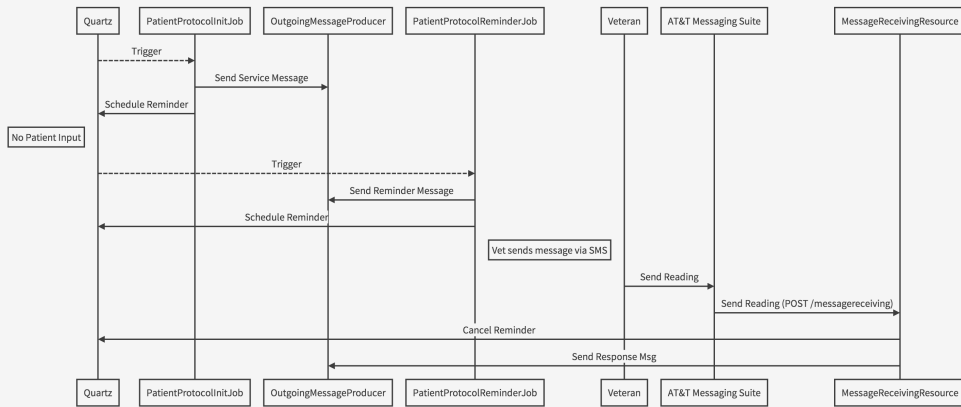
Send Message - Standard process for all messages generated by the system that are to be delivered to the Veteran. Outbound messages are put on the JMS queue. The queue consumer processes the message, checking the veteran preferences and delivery the message via the preferred mechanisms. Messages are always saved to the Mongo store to be available via the application.



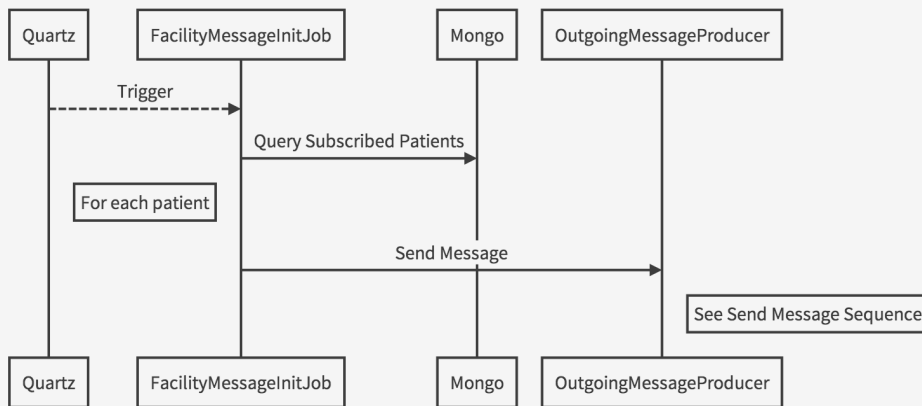
Receive Message - Standard process for all messages received via SMS. SMS Messages among all carriers are handled by AT&T Messaging Suite and forwarded to the system. The messages are immediately put on the JMS queue for processing. Messages are stored and processed for appropriate responses depending on message content.



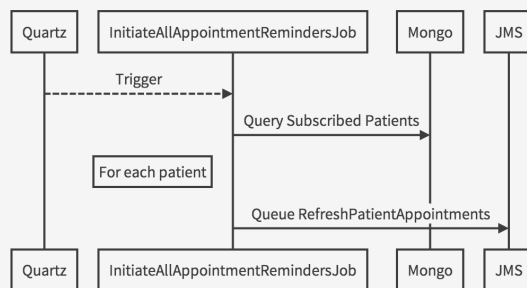
Protocol Message - Basic message flow for a protocol. The system initially sends a service message reminding the veteran to send their reading. A reminder message is scheduled and is delivered if the veteran does not send a reading. Once the Veteran sends the reading, the reminders are canceled and a response is sent back to the veteran.



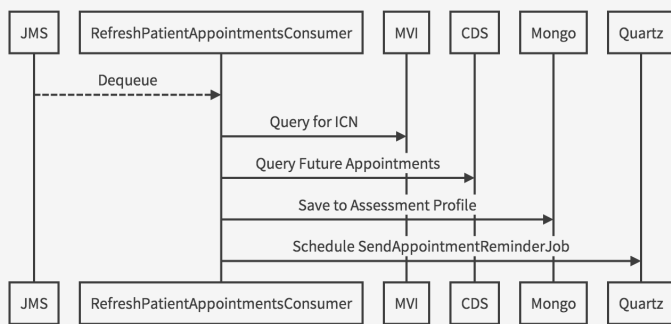
Broadcast Message Send



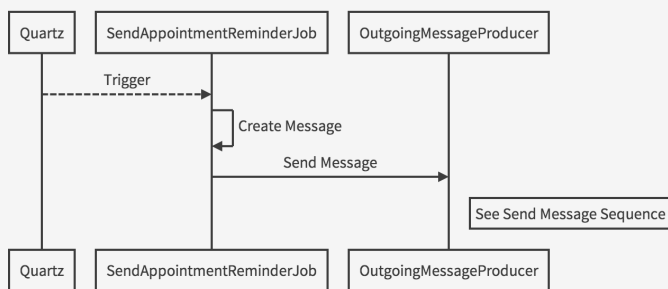
Appointment Reminders - Step 1 - A recurring job is used to scan all patients subscribed to appointment reminders. Each found patient results in a message being put on the queue to process that patient.



Appointment Reminders - Step 2 - Each patient is processed, querying CDS for future appointments. New scheduled jobs are created for the configured appointment reminder intervals.



Appointment Reminders - Step 3 - When time to send an appointment reminder, the Quartz job is triggered, and the message is sent to the OutboundMessage Producer.



Original Markdown source: [Annie-Sequence-diagrams.md](#)

Tooling: h

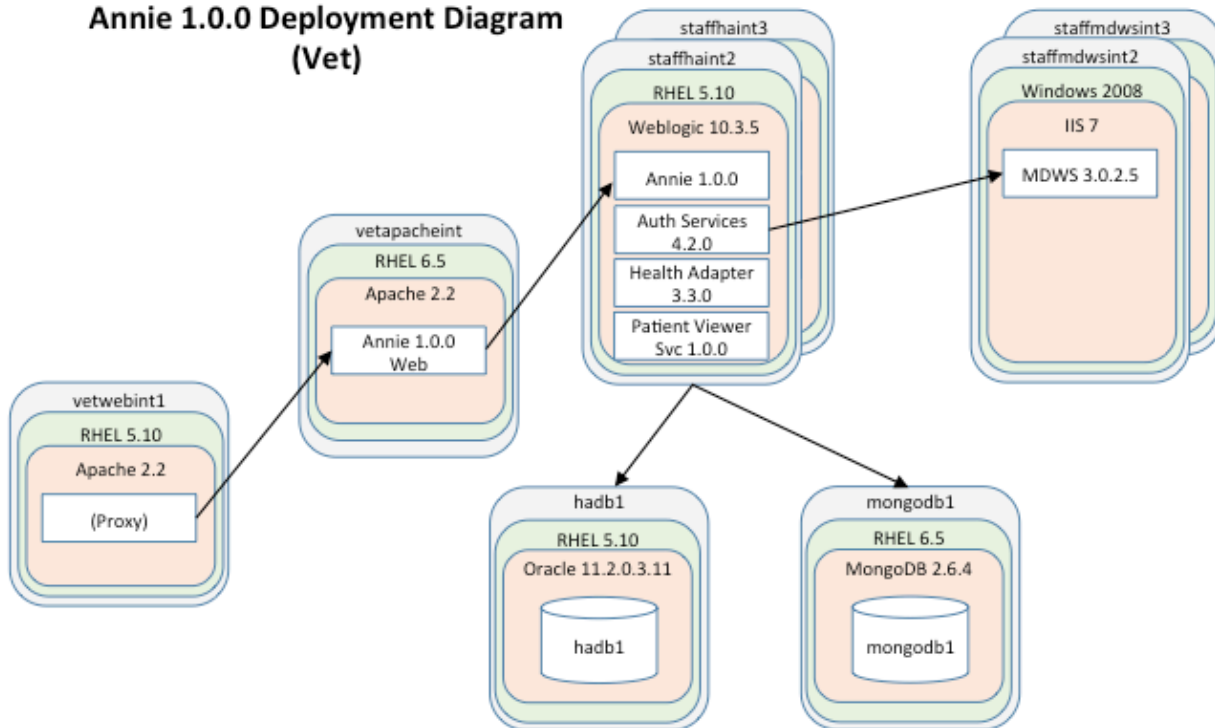
Class Diagrams

Classes relevant to CDS and MVI interactions

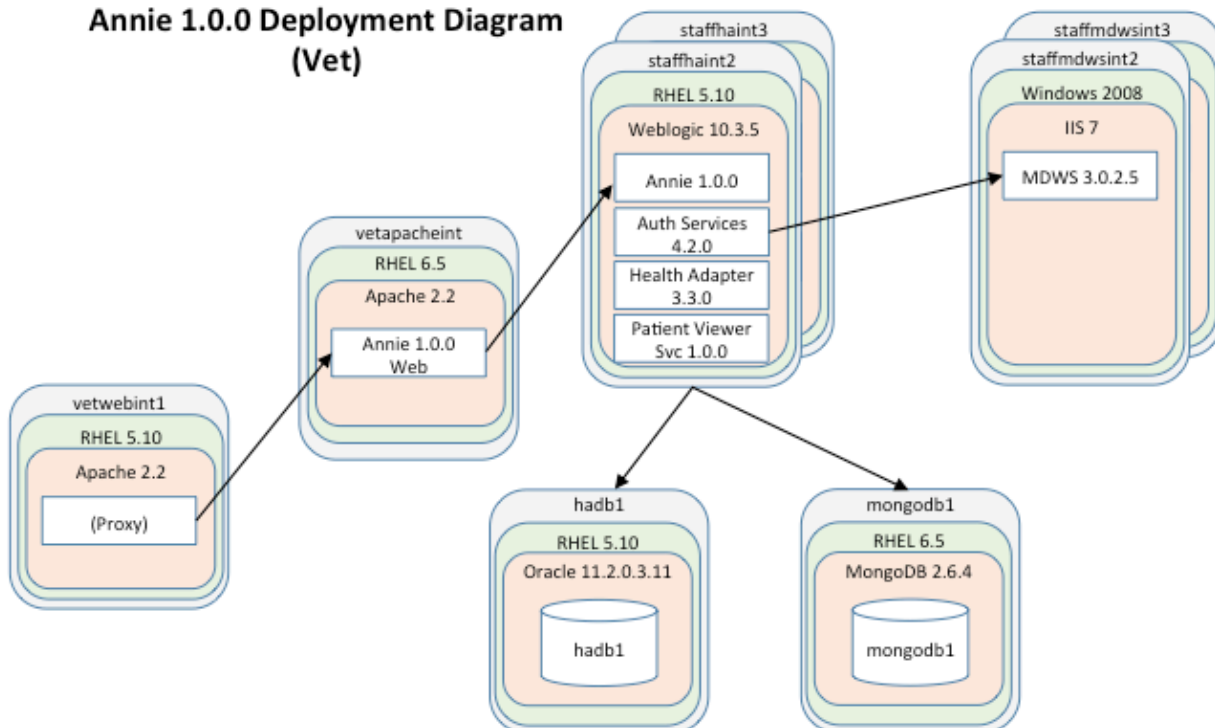
[AppointmentReminderSDDClassDiagram.pdf](#)

Deployment Diagrams

**Annie 1.0.0 Deployment Diagram
(Vet)**



**Annie 1.0.0 Deployment Diagram
(Vet)**





Text Commands

See information about [Text Commands](#)
(Includes OPT IN process)

ICD

Full documentation can be found at [ICD](#)

Device's OS	Native	Hybrid	Web-Only	OS version supported	Targeted devices (iPhone, iPad, Samsung model...)	Estimated Storage Required for Device
iOS			X	7.x	iPad iPhone (Veteran App only)	
Android			X		Smart Phones with Android OS (Veteran App only) Tablets with Android OS	
Windows 8			X		Desktop (w/IE 9)	

Technologies, Libraries, and Tools Used in the App	Version	On the TRM? (If not, provide a link to the waiver)
HTML 5	5	Yes
AngularJS	1.2.8	Yes
Highstock	1.3.10	No (Request being submitted - add to TRM)
MongoDB	2.6.5	Mongo DB Waivers:  
Java	6	Yes
Oracle	11	Yes
Weblogic	11	Yes
Quartz Scheduler	2.2	Yes

Data Collections

Ref #	NoSQL Collection	Description of data collection	Source of data	Structure
-------	------------------	--------------------------------	----------------	-----------

1	Courier.dictionary	The data dictionary to store predefined data, such as measurement, schedule type, available timezone.	SME, application requirement	<pre> { "_id" : ObjectId, "@class" : string, "value": string, "description": string, "name" : string, "readings" : array [{ "name" : string, "valid" : document { "low" : double, "high" : double }, "alert" : document { "low" : double, "high" :double }, "critical" : document { "low" : double, "high" : double }, "unit" : string }], "keywords" : array [string] } </pre>
---	--------------------	---	------------------------------	---

2	Courier.patients	Patients collection stores patient information and their preferences	VAMF, patient entered data	<pre> { "_id" : ObjectId, "ssn" : string, "gender" :string, "firstName" : string, "lastName" : string, "lastNameInitial" : string, "last4SSN" : string, "dob" : string, "patientExistInCms" : Boolean, "consent" : Boolean, "consentDate" : string, "facilityId" : null, "age" : Int32, "middleName": string, "preference" : document { "messageStarted" : Boolean, "messageStopped" : Boolean, "messagePaused" : Boolean, "phoneNumber" : string, "homePhoneNumber" : string, "messagingPreference" : string, "email" : string, "timeZone" : string, "phoneType" : string, "basicPhoneNonSecure" : Boolean, "receiveFacilityMessage" : Boolean, "receiveApptReminder" : Boolean }, "patientIdentifier" : document { "uniqueId" : string, "assigningAuthority" : string } } </pre>
3	Courier.protocolTemplates	Protocols collections stores all templates created by clinical admin	Clinical admin	<pre> { "_id" : ObjectId, "@class" : string, "status" : string, "type" : string, "name" : string, "description" : string, "serviceMessage" : string, "reminderMessage" : string, "confirmationMessage" : string, "lastMessageSent:: double, "service" : document { "_id" : ObjectId, "name" : string, "readings" : array [{ "name" : string, </pre>

```
"valid" : document
{
  "low" : double,
  "high" : double
},
"alert" : document
{
  "low" : double,
  "high" : double
},
"critical" : document
{
  "low" : double,
  "high" : double
},
"unit" : string
},
"firstLevelMessage" :
document
{
  "below" : string,
  "above" : string
},
"secondLevelMessage" :
document
{
  "below" : string,
  "above" : string
},
"criticalMessage" :
document
{
  "below" : string,
  "above" : string
},
"keywords" : array
[ string ],
"pattern" : string
},
"schedule" : document
{
  "acceptUnscheduledReading" : Boolean
  "scheduleType" :
document
{
    "@class" : string,
    "name" : string,
    "description" : string,
    "monday" : Boolean,
    "tuesday" : Boolean,
    "wednesday" : Boolean,
    "thursday" : Boolean,
    "friday" : Boolean,
    "saturday" : Boolean,
    "sunday" : Boolean
  },
  "scheduleTime" :
document
{
    "hour" : Int32,
    "min" : Int32,
    "meridian" : string,
    "description" : string
  },
  "isStartTimeRelative" :
Boolean,
  "relativeStartTime" :
document
{
    "value" : Int32,
```

```
"unit" : string
},
"isEndTimeRelative" :
Boolean,
"relativeEndTime" :
document
{
"value" : Int32,
"unit" : string
},
"reminderNumber" :
Int32,
"reminderInterval" :
document
{
"value" : Int32,
"unit" : string
},
"breach" : document
{
"repeatBreachValues" :
Boolean,
"breachesPerRepeatingA
lert" : Int32,
"breachesDays" : Int32,
"criticalPercentValues" :
Boolean,
"percentTriggersCriticalBr
each" : Int32
```


				<pre> } } } </pre>
	Courier.patientProtocol	patientProtocol stores protocols that have been assigned to patients by clinical staff	Clinical staff	<pre> { "_id" : ObjectId , "patientId" : string, "assignedDate" : string, "lastEditedDate" : string, "comments" : array [{ "content" : string, "user" : string, "time" : string, "date" : string, "changes" : array [{ "fieldName" : string, "oldValue" : string, "newValue" : string, "source" : string }] }], "status" : string, "protocol" : document { "status": boolean, "name" : string, "description" : string, "area" : string, "createDate" : string, "createBy" : string, "_id" string, "comments": array [], "templates" : array [{ "@class" : string, "status" : string, "type" : string, "name" : string, "description" :string, "serviceMessage" : string, "reminderMessage" :string, "confirmationMessage" : string, "service" : document { "_id" : string , "name" : string, "readings" : array [{ "name" : string, "valid" : document { "low" : double, "high" : double }, }, "alert" : document { "low" : double, "high" : double },], "critical" : document { </pre>

```
"low" : double,
"high" : double
},
"unit" : string
}],
"firstLevelMessage" :
document
{
"below" : string,
"above" : string
},
"secondLevelMessage" :
document
{
"below" : string,
"above" : string
},
"criticalMessage" :
document
{
"below" : string,
"above" : string
},
"keywords" : array
[string],
"pattern" : string
},
"schedule" : document
{
"acceptUnscheduledReading" :boolean
"scheduleType" :
document
{
"@class" : string,
"name" : string,
"monday" : boolean,
"tuesday" : boolean,
"wednesday" : boolean,
"thursday" : boolean,
"friday" : boolean,
"saturday" : boolean,
"sunday" : boolean
},
"scheduleTime" :
document
{
"hour" : Int32,
"min" : Int32,
"meridian": string,
"description" : string
},
"isStartTimeRelative" :
boolean,
"relativeStartTime" :
document
{
"value" : Int32,
"unit" : "string
"},
"isEndTimeRelative" :
boolean,
"relativeEndTime" :
document
{
"value" : Int32,
"unit" : string
},
"reminderNumber" :
Int32,
"reminderInterval"
:document
```

```
{
  "value" : Int32,
  "unit" : string
},
"breach" : document
{
  "repeatBreachValues" :
  boolean,
  "breachesPerRepeatingA
  lert" : Int32,
  "breachesDays" : Int32,
  "criticalPercentValues"
  :boolean,
  "percentTriggersCriticalBr
  each" : Int32
}
},
"_id" : string,
"startDate" : string,
"endDate" : string
```

				<pre> }} } } </pre>
	Courier.patientMessage	patientMessage keeps record of all messages received/sent to patients	System generated	<pre> { "_id" : ObjectId, "source" : string, "destination" : string, "text" : string, "type" : string, "networkSubmitTime" : string, "clientSubmitTime" : string, "serviceName": string, "adjustedSubmitTime" : string "msgDeliveryType" : array [string] "patientId" : string, "clientMessageId" : null, "costCentre" : null, "messageId" : null, "segmentCount" : double, "segmentNumber" : double, "referenceNumber" : double, "name" : null, "location" : null, "messages" : array [{ "_id" : null, "successCode" : string, "responseCode" : string, "responseString" : string, "messageId" : string }], "alertLevel": string, "timeZone": string } } </pre>
6	Courier.patientResponse ServiceData	patientResponseService Data stores patients response to difference vital services	System generated from patients response	<pre> { "_id" : ObjectId , "patientResponseReceiv edTime" : string, "patientId" : string, "alertLevel":string, "patientServiceReading" : document { "serviceName" : string, "serviceId" : string, "readings" : array [{ "name" : string, "value" : double, "unit" : string }] }, "patientResponseReceiv edTimeUTC" : string, "timezone": string } </pre>

7	Courier.staffRole	staffRole stores if the clinical staff has admin role	System admin	{ "_id" : ObjectId, "userIdentifier" : { "uniqueId" : string, "assigningAuthority" : string }, "roles" : array[string] }
8	Courier.facility	List of facilities available for Broadcast messages	System script	{ "_id" : string, "name" : string, "timezone" : string }
9	Courier.facilityMessages	Facility Broadcast Messages	BCM Admin	{ "_id" : ObjectId, "enabled" : Boolean, "name" : string, "assigningAuthority" : string, "createDate" : string, "createBy" : string, "message" : string, "startDate" : string, "endDate" : string, "createdDateUTC" : string, "schedule" : document { "scheduleType" : document { "@class" : string, "name" : string, "description" : string, "interval" : document { "value" : double, "unit" : string } }, "scheduleTime" : document { "hour" : double, "min" : double, "meridian" : string, "description" : string } }, "progress" : "Pending Delivery", "comments" : document { "content" : string, "user" : string, "time" : string, "date" : string, "changes" : array } }

10	Courier.patientPreference AuditTrailData	Audit Trail keeping a record of all preference changes made	System generated	{ "_id" : ObjectId, "content" : null, "user" : string, "time" : string, "date" : string, "changes" : array [{ "fieldName" : string, "oldValue" : string, "newValue" : string, }] }
11	Courier.assessmentProfile.futureAppointments	Current known future appointments for a Veteran.	CDS	{ "_id" : ObjectId, "facility" : string, "localStartDate" : string, "patientIcnIdentifier" : { "icnId" : string }, "patientEdipIdIdentifier" : { "edipId" : string }, "clinicPhone" : string }

Relational Structures

QUARTZ Schema is fully described in the following DDL script: [courier_dbinstall_r1.sql](#)

Table or Collection	Number of Rows / Entries	Estimated Size per Entry	Total Disk (minus overhead for indexes, etc)
Courier.facilityMessages		990 characters	
Courier.dictionary		5646 characters	
Courier.facility		93 characters	
Courier.patientResponseService Data		454 characters	
Courier.patients		420 characters	
Courier.protocolTemplates		2286 characters	
Courier.patientProtocol		3398 characters	
Courier.patientMessage		548 characters	
Courier.staffRole		166 characters	
Courier.patientPreferenceAuditT railData		724 characters	
Courier.assessmentProfile.future Appointments		234 characters	

Laws / regulations that direct the implementation of the app?

Not Applicable

Expected concurrent and maximum size of the user base?

See [Expected User Base After National Rollout](#)

User Category	Total Number of Users	Concurrent Users
Veterans		
Clinical Staff		
Clinical Admin		
Broadcast Admin		

For projected transaction counts, refer to the Top 10 Events on the ASSESS Form at [Annie_v1.0.0_ASSESS Form](#)

Interface Design

Existing VAMF Interfaces Used

(This section means VAMF services already deployed in production for this app)

Interface Name (incl. version number)	Data Used by Mobile App	Source of Data	Data Retrieved - provide specific data elements used by this app
Auth Services 4.0 (Veteran)	Authentication Response	DSLogon	First name, last name, EDIPI, auth token, date of birth, gender, ssn
Auth Services 4.0 (Staff)	Authentication Response	Link to Auth Services SDD	First name, last name, identifier, auth token
Patient Viewer Services 1.0	Patient Details	Link to PVS SDD	First name, last name, identifier, date of birth, gender, ssn PatientViewerServices/rest/patients?searchstring= PatientViewerServices/rest/patient/{assigningAuthority}/{uniqueid}/
CDS	Future Appointments	CDS/HDR	Start Date, Clinic Phone, Facility
MPI/MVI	Patient Correlations	MPI	Veteran ICN

New/Updated VAMF Interfaces Added

This section should contain services that are being deployed with this app - RESTful / other web services that become part of the VAMF service inventory, but are not available in production yet.

Interface Name (incl. version #)	Data Used by Mobile App	Source of Data	Included in Program-Level SDD?

External Interfaces Used

Interface Name	Data Used by Mobile App	Source of Data	Endpoint
AT&T Messaging Suite	Send SMS Messages to Mobile Numbers	Annie	████████████████████ ████████████████████ ██████████

For all services not currently in the Program-Level SDD, provide the documentation here that should be included - services, request data, response data, etc.

Dependencies/SLA

For example, you are reliant on a VA system to implement something for you, including supporting testing:

Not Applicable

Project Dependency	Point of Contact Information	Describe Dependency

Traceability

See Requirements Traceability Matrix: [Annie_v1.0.0_Requirements Traceability Matrix \(RTM\)](#)

EPIC	FEATURE	USER STORY	DESIGN COMPONENT

Developer & PM Contact Information

Contractor Developer Name/Point of Contact (POC)	VA E-Mail Address	Phone Number
██████	████████████████	██████
██████	████████████████	██████
██████	████████████████	██████
Developer Organization/Company	Contract Start Date	Contract End Date
Agilex	9/30/13	11/28/14
Web and Mobile Solutions PM or POC	VA E-Mail Address	Phone Number
██████████	████████████████	██████
VA Product Development PM or POC	VA E-Mail Address	Phone Number
██████████	V██████████████	██████