Virtual Assistant

Installation Instructions - VAEngine

Document Version v11.10

Date 21 December 2018

By:

[devang.vyas@creativevirtual.com](mailto:devang.vyas@creativevirtual.com)

**Distribution List**

|  |  |
| --- | --- |
| Sr. No. | Name |
|  | **Creative Virtual** |

**Document History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description of Change** | **Author** |
| 29-Sep-2009 | 0.1 | Initial draft. Creating installation instructions for VA Engine application. | Mayur Panchal, Sonal Quadros |
| 01-Oct-2009 | 1.0 | Reviewed and Updated | Avinash Gohil |
| 20-Oct-2009 | 1.1 | Added Section - Deployment | Mayur Panchal |
| 27-Oct-2009 | 1.2 | Reviewed and Updated | Avinash Gohil |
| 30-Oct-2009 | 1.3 | Incorporated review Comments from Thenappan (Test lead). | Mayur Panchal, Sonal Quadros |
| 02-Nov-2009 | 2.0 | Reviewed and Updated | Thenappan Palaniappan |
| 22-Nov-2009 | 2.1 | Updated java server configuration instructions and added configuration instruction for BOTSTATS functionality. | Bhushan Mhatre/ Sonal Quadros |
| 24-Nov-2009 | 2.2 | Reviewed and Updated | Avinash Gohil |
| 27-Nov-2009 | 3.0 | Reviewed | Thenappan Palaniappan |
| 3-Dec-2009 | 3.1 | Updated as per review comments received from Peter Behrend | Sonal Quadros/ Elizabeth Cherian |
| 8-Dec-2009 | 3.2 | Reviewed and Updated | Avinash Gohil |
| 14-Dec-2009 | 3.3 | Updated BotStats document | Bhushan Mhatre |
| 14-Dec-2009 | 4.0 | Modifying version to 4.0 as no comments were received regarding the document | William Lewis |
| 23-Mar-2010 | 4.1 | Modified program arguments for VAServer.exe | William Lewis |
| 23-Mar-2010 | 5.0 | Reviewed and Updated | Elizabeth Cherian |
| 15-Jul-2010 | 6.0 | Updated document Configuration Instruction VA-Engine document and Version updated to 6.0 | Cyril A. Gonsalves |
| 22-Jul-2010 | 7.0 | Updated document Configuration Instruction VA-Engine document and Version updated to 7.0 | Cyril A. Gonsalves |
| 26-Jul-2010 | 8.0 | Updated for change in the field « AcceptExternalSessionID » in the Engine configuration file. Also updated version to 8.0 | William Lewis |
| 30-Jul-2010 | 8.1 | Updated for insertion of new document Configuration Instructions VA-Engine - MS SQL.doc also modified to add new XML tag DatabaseType | Cyril A. Gonsalves |
| 30-Jul-2010 | 8.2 | Modified to incorporate review comments | Cyril A. Gonsalves |
| 02-Aug-2010 | 8.3 | Reviewed and Updated | William Lewis |
| 06-Sep-2010 | 9.0 | Changed version to 9.0. Also, modified for new nodes added to the Engine configuration file. | William Lewis |
| 10-Oct-2011 | 10.0 | Changed version to 10.0. Modified for new nodes added to the Engine configuration file as per Phase 2 modifications. | Roshni Samuel |
| 24-Mar-2014 | 10.1 | Modified for updated Configuration Instructions VA-Engine-BOTSTATS.doc and VirtualAssistant-VEngine\_Config.xml | Devang Vyas |
| 28-Mar-2014 | 10.2 | Modified for Python Interpreter pool size and SMTP connect timeout configuration parameters | Devang Vyas |
| 22-May-2014 | 10.3 | Modified for Timeout recognition configuration parameter | Devang Vyas |
| 18-Jun-2014 | 10.4 | Modified for Macro indexing configuration parameters | Devang Vyas |
| 11-Jul-2014 | 10.5 | Modified for cache support in standard UILayer | William Lewis |
| 28-Jul-2014 | 10.6 | Modified for Timeout recognition configuration changes | Devang Vyas |
| 20-Sep-2014 | 10.7 | Modified for Email on KB syntax error configuration | Devang Vyas |
| 03-Oct-2014 | 10.8 | Added option in UILayer to add module ID as a JVM argument | William Lewis |
| 23-Dec-2014 | 10.9 | Added x64 VC++ redistributable installation details. Also updated document format. | Devang Vyas, William Lewis |
| 28-Jul-2015 | 10.10 | Modified for Semantic search algorithm spelling threshold | Devang Vyas |
| 13-Aug-2015 | 10.11 | Modified for Python Interpreter pool related configurations | Devang Vyas |
| 05-Nov-2015 | 10.12 | Modified for Autocomplete related configuraiton | Devang Vyas |
| 25-Nov-2015 | 10.13 | Modified for Autocomplete ignore characters in language.xml | Devang Vyas |
| 22-Dec-2015 | 10.14 | Modified for ICU Library locale use in language.xml | Devang Vyas |
| 22-Dec-2015 | 10.15 | Reviewed and updated | William Lewis |
| 29-Feb-2015 | 10.16 | Modified for Language splitting configuration and ‘SplitUserInput’ and ‘DumpDictionary’ V-Engine configuration | Devang Vyas |
| 06-Jan-2017 | 10.17 | Added Python installation instructions for pre and post v2.2.1 V-Engine | Parvez Shaikh |
| 20-Apr-2017 | 10.18 | Modified for Maria DB ODBC driver and for Language configuration | Parvez Shaikh |
| 16-Aug-2017 | 10.19 | Modified for Candidate collection table name in VirtualAssistant-VEngine\_Config.xml and BOTSTATS document | Devang Vyas |
| 09-Oct-2017 | 11.0 | Added missing details of V-Engine configuration file | Devang Vyas |
| 01-Mar-2018 | 11.1 | Modified for including details related to UILayer (cv.war) v2.2.1.3 | Dhaval Mehta, William Lewis |
| 01-Mar-2018 | 11.2 | Reviewed and updated | William Lewis |
| 13-Mar-2018 | 11.3 | Modified for “EncryptedRequestResponse” V-Engine configuration | Parvez Shaikh |
| 14-Mar-2018 | 11.4 | Modified for “engine.request.encrypt” config and copy KBFileReadWrite.jar to tomcat/lib folder | Dhaval Mehta |
| 20-Mar-2018 | 11.5 | Reviewed and updated | William Lewis |
| 21-Mar-2018 | 11.6 | Modified for addition of Ubuntu configurations | Parvez Shaikh |
| 21-Mar-2018 | 11.7 | Reviewed and updated | Devang Vyas |
| 23-Aug-2018 | 11.8 | Modified for SubCategory Filtering configuration in VirtualAssistant-VEngine\_Config.xml | Devang Vyas |
| 23-Aug-2018 | 11.9 | Reviewed and updated | William Lewis |
| 20-Sep-2018 | 11.10 | Updated BOTSTATS document, cvutil.js, VAProperties.properties file, Removed SubCategory Filtering configuration in VirtualAssistant-VEngine\_Config.xml | Parvez Shaikh |

**Reference Documents**

|  |  |  |
| --- | --- | --- |
| **Reference ID** | **Document Name** | **Document Version & Date** |
| [R1] | CreativeVirtual\_P2- Build\_Instructions-VEngine\_Windows - 64 bit.docx | v7.3, 06-Jan-2017 |
| [R2] | Mail sent by William Perkins with subject “Re: VA Engine permissions” | 22-March-2010 |
| [R3] | CreativeVirtual - VPT Log Level Requirements.docx | v1.0, 24-Mar-2014 |

**Abbreviations**

|  |  |
| --- | --- |
| VA | Virtual Assistant |
| ACE | Adaptive Communication Environment |
| TAO | The Ace ORB |
| ICU | International Components for Unicode |

The statements in red refer to statements that have been modified from the previous version.

Table of contents

[1 Introduction 7](#_Toc522791799)

[1.1 Purpose 7](#_Toc522791800)

[1.2 Scope 7](#_Toc522791801)

[2 Assumptions 8](#_Toc522791802)

[2.1 Pre-Requisites 8](#_Toc522791803)

[3 Configuration required on windows machine to connect to UBUNTU machine 10](#_Toc522791804)

[4 Deployment 14](#_Toc522791805)

[4.1 Web Server 14](#_Toc522791806)

[4.2 VA-Engine 14](#_Toc522791807)

[4.3 VA-EngineFactory 15](#_Toc522791808)

[4.4 Install Microsoft Visual C++ 2008 Redistributable Package 15](#_Toc522791809)

[4.5 Install Python 16](#_Toc522791810)

[4.6 Security related DLLs 17](#_Toc522791811)

[5 RUNNING the c++ engine application 18](#_Toc522791812)

[5.1 Naming Service 18](#_Toc522791813)

[5.2 VA Engine Factory 20](#_Toc522791814)

[5.3 VA Server 22](#_Toc522791815)

[6 RUNNING the java web-interface server 46](#_Toc522791816)

[6.1 Setup UILayer (cv.war) in Tomcat 46](#_Toc522791817)

[6.1.1 Configuration Setup 46](#_Toc522791818)

[6.2 Run the Web Application 54](#_Toc522791819)

[6.3 Run the Administrator Application 56](#_Toc522791820)

[6.4 Setup for SSO for UILayer using OpenAM 57](#_Toc522791821)

[6.5 Tomcat Security Issues & Resolutions (Optional) 58](#_Toc522791822)

TABLE of Figures

[Figure 1: putty configuration 10](#_Toc522791823)

[Figure 2: putty window 11](#_Toc522791824)

[Figure 3: Putty Reconfiguration window 12](#_Toc522791825)

[Figure 4: Site Manager 13](#_Toc522791826)

[Figure 5: Windows Engine Deployment structure 14](#_Toc522791827)

[Figure 6: Ubuntu Engine Deployment structure 15](#_Toc522791828)

[Figure 7: Windows Engine Factory Deployment structure 15](#_Toc522791829)

[Figure 8: Ubuntu Engine Factory Deployment structure 15](#_Toc522791830)

[Figure 9 **:** Naming Service Running Screen 18](#_Toc522791831)

[Figure 10 **:** VA Engine Factory Running Screen 20](#_Toc522791832)

[Figure 11 **:** VA Engine Factory Running Screen 21](#_Toc522791833)

[Figure 12 **:** VA Server Welcome Screen 23](#_Toc522791834)

[Figure 13: Putty screen 24](#_Toc522791835)

[Figure 14 : Password Generation 53](#_Toc522791836)

[Figure 15: Start VA 55](#_Toc522791837)

[Figure 16: Chat with VA Avatar 55](#_Toc522791838)

[Figure 17: Administrator Login Screen 56](#_Toc522791839)

[Figure 18: Authentication Details 57](#_Toc522791840)

[Figure 19: Administrator Details 57](#_Toc522791841)

TABLES

[Table 1: Nodes of “EngineFactory.xml” 22](#_Toc522791842)

[Table 2:Nodes of “VirtualAssistant-VEngine\_Config.xml” 40](#_Toc522791843)

[Table 3: Nodes of “language.xml” 45](#_Toc522791844)

[Table 4: Nodes of “Web\_Config.xml” 48](#_Toc522791845)

[Table 5: Nodes of “web.xml” 49](#_Toc522791846)

[Table 6: Nodes of “tomcat-users.xml” 52](#_Toc522791847)

# Introduction

The VA-Engine application being developed for Creative Virtual consists of the following 2 major components:

* **Web-interface server**

The Web-interface-server has been developed in JAVA and can be hosted on the Tomcat Server version 6.0.18+.

* **Engine Application**

The Engine application has been developed in Visual Studio 2008(Windows)/G++ 4.4.7(Ubuntu). The Engine application makes use of the ACE TAO, ICU and Python libraries.

## Purpose

This document provides step-by-step instructions on how to run the VA-Engine application (JAVA and C++).

## Scope

The scope of this document is to provide instructions on how to configure and run the V-Engine application. It does not provide information about building the application. The scope is limited to run the application on a machine with Windows and Ubuntu Operating system.

# Assumptions

## Pre-Requisites

The following are the pre-requisite’s to run the **Application**:

1. Application is successfully built using instructions given in [[R1](#R1)] OR the release deployables are available.
2. Install **Java 1.5+, JacORB 2.3.1, Tomcat 6.0.18+** as mentioned in [[R1](#R1)].
3. The following System variables along with their values need to be added based on the versions/paths installed. If already present, existing variables should be edited by adding these values using “;” as a separator.

|  |  |
| --- | --- |
| System Variable Name | Value |
| JAVA\_HOME | C:\Program Files\Java\jdk1.5.0\_14 |
| TOMCAT\_HOME | D:\apache-tomcat-6.0.18 |
| PATH | %JAVA\_HOME%\bin;%TOMCAT\_HOME%\bin |

1. To change Tomcat’s default ORB to JacORB, the following needs to be incorporated.
2. Create a directory named endorsed in the tomcat installation folder

e.g. D:\apache-tomcat-6.0.18\endorsed

1. Add all the jar files present in the lib directory of the JacORB 2.3.1 download in this “endorsed” directory.
2. Install Python. Check details in [Section 4.5](#_Install_Python).
3. Install external Python scripting modules, if required, for e.g. pyodbc
4. Based on the database being used, configure BOTSTATS functionality by using any one of the below documents.

For using MYSQL database

1. Windows

Configuration Instructions VA-Engine-BOTSTATS.doc



1. Ubuntu

Configuration Instructions VA-Engine-BOTSTATS on Ubuntu.doc



1. For Ubuntu, **Use following steps to download “putty.exe” --🡪 done till here**
   1. **Go to link** [putty.exe](http://the.earth.li/~sgtatham/putty/latest/x86/putty.exe) to download putty.exe.
2. For Ubuntu**, Download “FileZilla” setup from following link**

[**https://www.ohloh.net/p/filezilla/download?filename=FileZilla\_3.3.1\_win32-setup.exe**](https://www.ohloh.net/p/filezilla/download?filename=FileZilla_3.3.1_win32-setup.exe)

1. For Ubuntu, **make** and **gcc/g++ 4.4.7** are required to build third party libraries.

Ubuntu 17.10 Server will get/have updated gcc/g++ version which will not work.

Refer section 3 in following document to downgrade/install required gcc/g++ version.



**Please note that the application will not run without the above mentioned pre-requisites.**

# Configuration required on windows machine to connect to UBUNTU machine

The C++ Engine workspace consists of the following components:

1. **Putty Configuration**

* Double click on “putty.exe”.

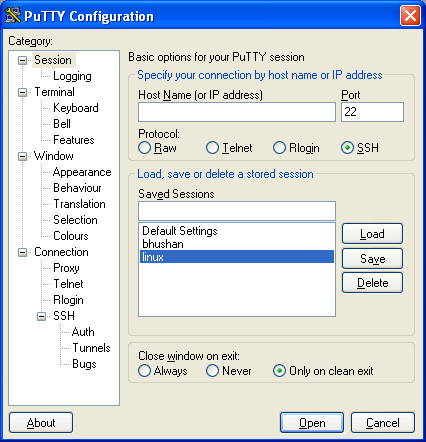


Figure 1: putty configuration

* Select **“SSH”** radio button. Please refer figure 1.
* Type host name or IP address of the system you want to connect (E.g. 172.16.220.224).
* You can save session by using “Save” button (Refer figure 1). When you login next time you can directly click on the session you created with configuration mentioned in above steps.
* Click on open. Putty window will be pop up. (Refer Figure 2).



Figure 2: putty window

* Enter Login name as “test1” and enter password (Please get password from concern person (project manager)).
* Please note that putty window output can be redirected to text file by using following steps
  + Right click on the task bar.
  + Click on “Change settings..”.

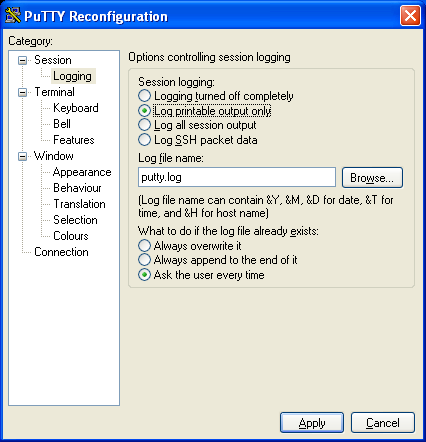


Figure 3: Putty Reconfiguration window

* + Select “Session->Logging” from “Category”. (Refer above figure) and then select radio button “Log Printable output only” from “Options controlling session logging->Session logging”.
  + Put log file name in input box named “Log file name:”, you can change default path by using Browse button.
* You can use this putty window to build application by using steps mentioned in [section 3](#_Building_the_c++_engine application).

1. **FileZilla setup**

Double click on “FileZilla” setup exe and follow installation instruction

1. **Connect FileZilla to LINUX machine using following steps**

* Go to Start->All programs->FileZilla FTP client. You will get “FileZilla” window.
* Use CTRL+S. Refer Figure 4.
* Select General tab.

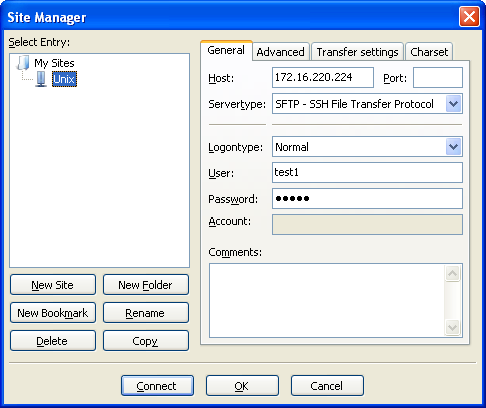


Figure 4: Site Manager

* Put hostname “172.16.220.224”. Select Servertype:, Logontype:, User: as shown in figure 3. Please get password from concern person (project manager).
* Click on “Connect” button.

# Deployment

## Web Server

To deploy the web server, add the UILayer war file (cv.war) to the webapps directory present in the Tomcat installation directory.

E.g. D:\apache-tomcat-6.0.18\webapps\cv.war

**Module ID as part of JVM Arguments**

Module ID can now also be added as part of the JVM arguments when Tomcat is started. This can be done by modifying the ‘Catalina.bat’ file which may be found in the ‘bin’ folder of the Tomcat installation. A key-value pair ‘-DClientDetails=MODULE\_ID’ needs to be appended to the existing ‘JAVA\_OPTS’ property in the following manner:

set JAVA\_OPTS=%JAVA\_OPTS% %LOGGING\_CONFIG% -DClientDetails=BANK

In this case, the module id ‘BANK’ will be used by the UILayer to reference the Engine. If not JVM argument is found then it will look for the Module ID as usual in the web.xml file.

**Copy KBFileReadWrite.jar from release package to TOMCAT\_HOME\lib folder.**

**Cache support in the UILayer (Please note, this has not been tested with v2.2.1.4 UILayer)**

Caching support in the UILayer has been disabled by default.

To enable caching the following steps need to be followed:

1. Deploy the war file (eg. ‘cv.war’) into tomcat server and start the server, so that the war file is blown up inside the ‘webapps’ folder
2. Next navigate to the folder ‘cv/WEB-INF’ and modify the ‘web.xml’ to include the caching filter and its mapping. Please note in the delivered war file the caching code is present within the web.xml file but is commented. Please uncomment the 2 code snippets beginning with ‘Uncomment/Insert Following code for Caching Starts’
3. Please note, Tomcat will have to be restarted after making changes to web.xml to take the desired effect

## VA-Engine

1. **Windows**

Deploy the VAEngine\_Deploy package on D: drive of the production server. The directory structure should look like below:

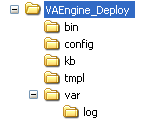


Figure 5: Windows Engine Deployment structure

1. **Ubuntu**

Deploy the VAEngine\_Deploy package on $HOME of the production server. The directory structure should look like below:

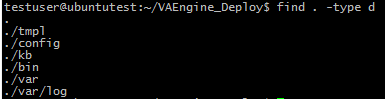


Figure 6: Ubuntu Engine Deployment structure

## VA-EngineFactory

1. **Windows**

Deploy the VAEngineFactory\_Deploy package on D: drive of the production server. The directory structure should look like below:



Figure 7: Windows Engine Factory Deployment structure

1. **Ubuntu**

Deploy the VAEngineFactory\_Deploy package on $HOME of the production server. The directory structure should look like below:



Figure 8: Ubuntu Engine Factory Deployment structure

## Install Microsoft Visual C++ 2008 Redistributable Package

For 64-Bit engine, install Microsoft Visual C++ 2008 redistributable package (x64) where V-Engine is deployed if not installed.

The Microsoft Visual C++ 2008 Redistributable Package (x64) installs runtime components of Visual C++ Libraries required to run 64-bit applications developed with Visual C++ on a computer that does not have Visual C++ 2008 installed.

The package can be downloaded from following link,

<http://download.microsoft.com/download/d/2/4/d242c3fb-da5a-4542-ad66-f9661d0a8d19/vcredist_x64.exe>

## Install Python

Python version is switched from 2.6.x to 2.7 from V-Engine 2.2.1. All earlier V-Engine releases should use Python v2.6.x.

Based on the architecture, download Python package from the following location and install. Please refer section 4.3 in [[R1](#R1)].

1. Pre v2.2.1 V-Engine
2. 32 bit

<https://www.python.org/ftp/python/2.6.2/python-2.6.2.msi>

1. 64 bit

<https://www.python.org/ftp/python/2.6.2/python-2.6.2.amd64.msi>

1. Ubuntu
2. **Python version 2.6.2 named “Gzipped source tar ball (2.6.2) (sig)”** should be **downloaded** from the following location: <https://www.python.org/download/releases/2.6.2>
3. Use putty to login in Ubuntu system. Go to directory where tar archive file is located.

Then extract “**Python-2.6.2.tgz**” tar file using following command.

**$** tar -xvf **Python-2.6.2.tgz –C $HOME**

A new folder should be at the path “**$HOME*/*Python-2.6.2**” will be created and the

Code will be extracted to it.

1. Change directory using command “**$ cd $HOME/Python-2.6.2**”
2. V-Engine v2.2.1 onwards
3. 32 bit

<https://www.python.org/ftp/python/2.7.12/python-2.7.12.msi>

1. 64 bit

<https://www.python.org/ftp/python/2.7.12/python-2.7.12.amd64.msi>

1. Ubuntu
2. **Python version 2.7.12 named “Gzipped source tar ball”** should be **downloaded** from the following location: <https://www.python.org/downloads/release/python-2712>
3. Use putty to login Ubuntu system. Go to directory where tar archive file is located.

Then extract “**Python-2.7.12.tgz**” tar file using following command.

**$** tar –xvf **Python-2.7.12.tgz**

A new folder should be at the path “**$HOME*/*Python-2.7.12**” will be created and the

code will be extracted to it.

1. Change directory using command “**$ cd $HOME/Python-2.7.12**”

For Ubuntu, proceed with the following steps:

1. Run Command "**$** ./configure --enable-shared", which determines your system configuration and creates the Makefile. (It takes a minute or two -- please be patient!)
2. To build Python, use following command:

**$ make**

1. To install Python, use following command:

**$ make install**

**Please Note:** Administrator rights are required for installation.

## Security related DLLs

Copy the OS-specific (32-bit / 64-bit) DLL files - **KBFileReadWrite.dll** & **ACE.dll** from the release directory in the release package to the “system32” folder under Windows directory.

# RUNNING the c++ engine application

Following applications are required to communicate between the web interface (Tomcat) and the C++ engine.

1. Naming Service (Naming\_Service.exe)
2. VA Engine Factory (VAEngineFactory.exe)
3. VA Server (VAServer.exe)

## Naming Service

1. **Windows**

Perform following steps to start the Name Service:

1. Start command prompt & change directory to *VAEngine\_deploy\bin*
2. Start Naming Service using below command:
   1. Naming\_Service.exe -m 0 -ORBEndPoint iiop://<Machine IP>:<Port No>

Where <Machine IP> - IP address of local host

<Port No> - Port Number

e.g.: Naming\_Service.exe -m 0 -ORBEndPoint iiop://172.16.232.179:9990

On success, Naming Service will start with a blinking cursor as shown in the image below:

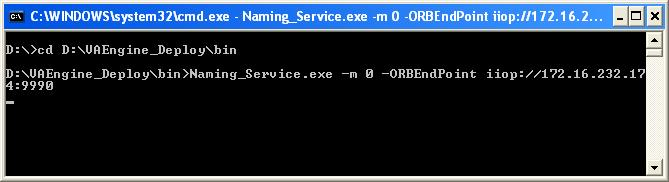


Figure 9 **:** Naming Service Running Screen

* 1. **Ubuntu**

1. Start new putty window with login name and password & change directory to *VAEngine\_deploy/bin*. Copy Naming Service binaries and third party libraries from release package to *VAEngine\_deploy/bin*

Set following environment variable using putty window.

“export LD\_LIBRARY\_PATH=/usr/local/lib:$LD\_LIBRARY\_PATH”

“export LD\_LIBRARY\_PATH=$HOME/*VAEngine\_deploy/bin*:$LD\_LIBRARY\_PATH”

1. Execute following command

“./Naming\_Service -m 0 -ORBEndPoint iiop://172.16.220.224:9990”

1. If command is successful, naming service will run without an error.
2. If exception occurs check if “Naming\_Service” is already running on the Ubuntu machine by using following command which gives you list of all running processes with their respective process ID.

“ps -A”

1. If “Naming\_Service” process is already running use following command to kill the process where PID is the process id of the process.

“kill PID”

E.g. if PID is 7000 then command would be “kill 7000” and then press enter.

1. Try starting naming service again.

## VA Engine Factory

1. **Windows**

Perform following steps to start the VA Engine Factory:

1. Edit *“*VAEngine\_Deploy\*bin\VAServer.bat”* file to provide correct naming service detail.

E.g.: The sample VAServer.bat is as shown below:

*cd \*

*d:*

*cd “d:\BT\bin”*

*VAServer.exe –ORBInitRef NameService=corbaloc:iiop:172.16.232.178:9990/NameService*

**Please note: “VAServer.exe” name and IP address must be correct. If it is wrong, VA Engine factory will restart executable mentioned in “VAServer.bat” file recursively.**

1. Open command prompt & change directory to “VAEngineFactory\_Deploy\*bin*” directory.
2. Start VA Engine Factory using command,
   1. VAEngineFactory.exe "-ORBInitRef NameService=corbaloc:iiop:<Machine IP>:<Port No>/NameService"

Where <Machine IP> - IP address on which Name Service is running

<Port No> - Name Service Port Number

E.g. VAEngineFactory.exe "-ORBInitRef NameService=corbaloc:iiop:172.16.232.179:9990/NameService"

On success VA Engine Factory will start with blinking cursor as shown in the image below:

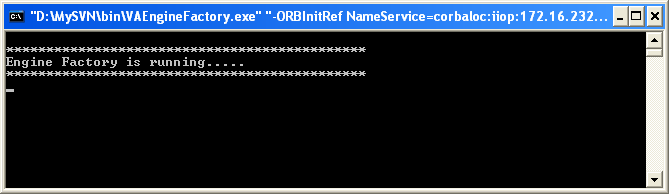


Figure 10 **:** VA Engine Factory Running Screen

* 1. **Ubuntu**

1. Start new putty window with login name and password & change directory to *VAEngineFactory\_Deploy/bin*. Copy EngineFactory binaries and third party libraries from release package to *VAEngineFactory\_deploy/bin*

Set following environment variable using putty window.

“export LD\_LIBRARY\_PATH=usr/local/lib:$LD\_LIBRARY\_PATH”

“export LD\_LIBRARY\_PATH=$HOME/*VAEngineFactory\_deploy/bin*:$LD\_LIBRARY\_PATH”

1. Use following command to run “EngineFactory”. Command is for the name service with IPAddress **172.16.220.224 and port number 9990.**

**“./EngineFactory -ORBInitRef NameService=corbaloc:iiop:172.16.220.224:9990/NameService”**

1. If “EngineFactory” is started successfully following message will get displayed on the putty screen.



Figure 11 **:** VA Engine Factory Running Screen

On execution, VA Engine Factory extracts values from “*EngineFactory.xml*” configuration file and initializes itself with them. The configuration file is present in “*config*” folder.

|  |  |  |
| --- | --- | --- |
| Node Name | Attribute Name | Description |
| <Configuration> | - | This is main node of Engine Factory configuration file name. |
| <Configuration/Moduleinfo> | - | This would contain the Engine Factory POA and Naming Service context name. **(Please note that POA and Naming service context name in EngineFactory.xml must be same as the respective VAEngine application POA and Naming service context name. If POA and Naming service context name is not same then VAEngine factory will restart executable mentioned in “VAServer.bat” file recursively. )** |
| <Configuration/Modules> | - | This would contain list of VA Engines information. VA Engine information would contain Naming Service context name, Batch file name and Module load time. (Module load time is time in seconds, required to start the “VAServer.exe”) |

Table 1: Nodes of “EngineFactory.xml”

Sample “EngineFactory.xml” for single VAServer is given below:



Sample “EngineFactory.xml” for multiple VAServers are given below:



## VA Server

**Please Note: the VAServer application requires a license file (License.lic) to run on any particular machine. The license file may be generated using the License Generation utility which comes along with the V-Builder source code. The steps to build the license file may be found in “VirtualAssistant-BuildAndInstallationInstructions-VBuilder.doc”**

1. **Windows**

Perform following steps to start the VA Server:

1. Open command prompt & change directory to “VAEngine\_Deploy\*bin*” directory.
2. Start VA Engine Server using command,
3. VAServer.exe "-ORBInitRef NameService=corbaloc:iiop:<Machine IP>:<Port No>/NameService [--log-error=LogFileDirectory]”

Where <Machine IP> - IP address on which Name Service is running

<Port No> - Name Service Port Number

[--log-error=LogFileDirectory] – Optional parameter to configure the directory paths of the various Log files, namely, “error.txt”, “VAEngineLogger.txt” and “VA\_Server.txt”. This is based on method proposed and accepted in [[R2]](#R2).

It is important to note here that, in case the “LogFileDirectory” path does not have adequate “write” rights, then the log files will be attempted to be written into the “var\log” folder. If sufficient rights are not present in the “var\log” folder, then the “bin” folder will be used to write the log files. In case the “bin” folder does not have “write” rights then the VAServer.exe will not start.

If the [--log-error=LogFileDirectory] argument is not given as a program argument, then by default the “var\log” folder will be used and if sufficient “write” rights are not available then the “bin” folder would be used to write the above mentioned Log files. In case the “bin” folder does not have “write” rights then the VAServer.exe will not start.

Also, if there is an error in the [--log-error=LogFileDirectory] argument, then the VAServer.exe will not start, indicating an error in the program arguments.

E.g. *VAServer.exe –ORBInitRef NameService=corbaloc:iiop:172.16.232.179:9990/NameService --log-error=D:\CreativeVirtual\VirtualAssistant\logs*

On execution VA Server extracts Knowledgebase name and other initialization values from “*VirtualAssistant-VEngine\_Config.xml*” configuration file. The configuration file is present in “*config*” folder. It also reads “language.xml” to initialize language module. The configuration files description is given below. If everything is correct VA Server then it prints Knowledgebase information and welcome message as shown below image. On error it displays error message on screen. The detailed error is logged into “*D:\CreativeVirtual\VirtualAssistant\logs\error.txt*” file. As mentioned above if sufficient “write” rights are not available in the folder “*D:\CreativeVirtual\VirtualAssistant\logs”* then the file will be written to “var\log\error.txt” and in case of failure here, then into the “bin” folder itself. The VAServer.exe will not start if sufficient rights are not available on the “bin” folder.

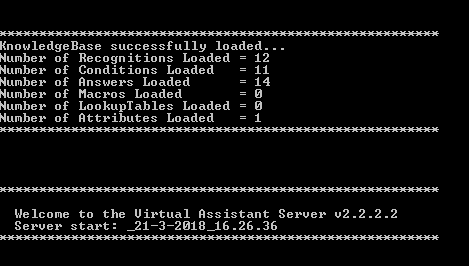


Figure 12 **:** VA Server Welcome Screen

1. **Ubuntu**
2. Start new putty window with login name and password & change directory to *VAEngine\_Deploy/bin*. Copy VAEngine binaries and third party libraries from release package into *VAEngine\_Deploy/bin*

Set following environment variable using putty window.

“export LD\_LIBRARY\_PATH=/usr/local/lib:$LD\_LIBRARY\_PATH”

“export LD\_LIBRARY\_PATH=$HOME/VAEngine\_Deploy/bin:$LD\_LIBRARY\_PATH”

1. Use following command to run the “EngineServer”. Command is for the name service with IPAddress **172.16.220.224 and port number 9990.**

**“./EngineServer -ORBInitRef NameService=corbaloc:iiop:172.16.220.224:9990/NameService”**

1. If “EngineServer” is started successfully following message will get displayed on the putty screen as shown in figure 4.

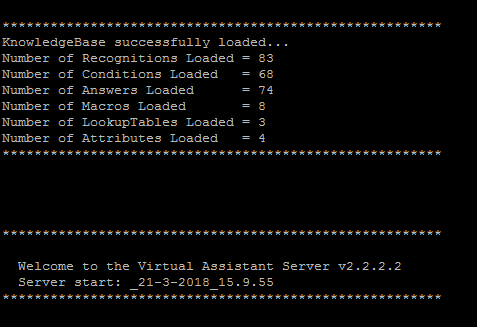


Figure 13: Putty screen

Following table describes important nodes of “VirtualAssistant-VEngine\_Config.xml” configuration file:

|  |  |  |
| --- | --- | --- |
| Node Name | Attribute Name | Description |
| <Engine> | buildnumber  releasedate  botname  version | This is main node of engine configuration file name. Attributes indicates Build number, Release date, Bot Name and version number of VEngine. |
| <ServerInfo/ KnowledgebaseFilePath> | - | This would contain the Knowledgebase file path.  E.g.  ../kb/KnowledgeBase.vpx |
| <ServerInfo/PublishFTPDirPath> | - | This would contain folder name in which Published files are located.  E.g.  ../VAEngineFTPFiles/ |
| <ServerInfo/BackupAutoCompleteDirPath> | - | This would contain folder name in which Old autocomplete files will be backed up.  E.g.  ../Backup/ |
| <ServerInfo/BackupEmailAttachmentsDirPath> | - | This would contain folder name in which Old attachment files will be backed up.  E.g.  ../Backup/ |
| <ServerInfo/BackupKBDirPath> | - | This would contain folder name in which Old knowledgebase files will be backed up.  E.g.  ../Backup/ |
| <ServerInfo/ AutoCorrectionFilePath> | - | This would contain the Auto Correction file path.  E.g.  ../kb/auto\_corrections.txt |
| <ServerInfo/ AbbreviationFilePath> | - | This would contain the Abbreviation file path.  E.g.  ../kb/abbrevations.txt |
| <ServerInfo/ TableAnswerFilePath> | - | This would contain the folder name in which Table Answer files are located.  E.g. ../kb/ |
| <ServerInfo /TemplateDirectory> | - | This would contain the folder name in which Template files are located.  E.g. ../tmpl/ |
| <ServerInfo /TriggerLogFilePath> | - | This would contain the Trigger log file path.  E.g.  ../var/log/Trigger.log |
| <ServerInfo /AdminLogFilePath> | - | This would contain the Admin log file path.  E.g.  ../var/log/admin.log |
| <ServerInfo /TranscriptFilePath> | loglevel | This would contain the folder name in which Transcript files are located. Attribute “loglevel” indicates transcript file log level. Possible values for this attribute are off, all, performance. Please refer [[R3]](#R3) for details.  E.g. ../var/log/ |
| <ServerInfo /SpecialLogDirectory> | - | This would contain the folder name in which Special Log files are located.  E.g. ../var/log/ |
| <ServerInfo/ClearUserInput> | - | This would contain true/false value indicating whether to clear user input or not. The default value is true. |
| <ServerInfo/NumberOfSubmits> | - | This would contain positive value for Number of submits. The default value is 60. |
| <Templates> | - | This would contain list of the templates. The “STANDARD” template is compulsory and template name should not repeat else engine would not start. |
| <Emotions> | template | This would contain the list of emotions. The “NEUTRAL” emotion is compulsory and emotion name should not repeat else engine would not start. |
| <ConnectionInfo/ ORBThreadPoolSize> | - | This would contain number of ORB thread pool size. The default value is 1. |
| <ConnectionInfo/ PythonInterpreterPoolSize> | - | This would contain number of python interpreters that engine will create in python interpreter pool. ORBThreadPool size must be less than or equal to Python Interpreter Pool size. |
| <ConnectionInfo/ PythonInterpreterRetry> | - | This would contain retry count to get python interpreter if request fails to get python interpreter from python interpreter pool. Retry count should be configured in such way that total retry interval (retry count \* retry interval) is greater than or equal to “VEngineScriptTimeOut”. If not than engine will use configuration that satisfies above mentioned condition. |
| <ConnectionInfo/ PythonInterpreterRetryInterval> | - | This would contain wait/sleep time interval in milliseconds for get python interpreter retry. Retry interval should be configured in such way that total retry interval (retry count \* retry interval) is greater than or equal to “VEngineScriptTimeOut”. If not than engine will use configuration that satisfies above mentioned condition. |
| <ConnectionInfo/ Servants> | type | This would contain the list of Servant. |
| <ConnectionInfo/ Servants/Servant> | name  nscontext  poaname  configfile | Please provide correct dll name. i.e. For release Servant should be name="VAEngine.dll" and for debug it should be name="VAEngined.dll" else engine will not behave as expected.  Naming service context. It should match to values configured in EngineFactory.xml  Portable object adapter name. It should match to Module Id configured in EngineFactory.xml  This would contain file path of V-Engine configuration file |
| <Email> | - | This would contain the valid SMTP server data and email address else email would not be sent. |
| <Email><SMTPServer> | - | This would contain IP address or Host name of SMTP server |
| <Email><SMTPPort> | - | This would contain SMTP Port Number |
| <Email><SMTPFrom> | - | This would contain SMTP 'From' email Id |
| <Email><SMTPReplyTo> | - | This would contain SMTP 'Reply To' email Id |
| <Email><SMTPIdent> |  | This would contain SMTP Ident to be used |
| <Email><ConnectionTimeout> | - | Value for <ConnectionTimeout> would contain SMTP server connect timeout value in seconds. Default value is set to 60 seconds. |
| <Email><EmailOnKBSyntaxErrors> | - | Value for <EmailOnKBSyntaxErrors> would indicated whether to send e-mail on KB syntax error (build condition tree failed for normal condition or table answer condition) or not. Possible values “on” or “off” |
| <FailureAlertMailReceivers> | - | This would contain failure mail receiver details |
| <FailureAlertMailReceivers><DSNFailure> | - | This would contain DSN failure mail receiver details |
| <FailureAlertMailReceivers><DSNFailure><Name> | - | This would contain DSN failure alert receiver name |
| <FailureAlertMailReceivers><DSNFailure><EmailId> | - | This would contain DSN failure alert receiver email Id |
| <FailureAlertMailReceivers><EngineFailure> | - | This would contain DSN failure mail receiver details |
| <FailureAlertMailReceivers><EngineFailure><Name> | - | This would contain Engine failure alert receiver name |
| <FailureAlertMailReceivers><EngineFailure><EmailId> | - | This would contain Engine failure alert receiver email Id |
| <ClientEmail> |  | This would contain the valid SMTP server data and email address of client else email would not be sent. |
| <ClientEmail><SMTPPort> | - | This would contain Client SMTP Port Number |
| <ClientEmail><SMTPFrom> | - | This would contain Client SMTP From email Id |
| <ClientEmail><SMTPReplyTo> | - | This would contain Client SMTP Reply To email Id |
| <ClientEmail><SMTPIdent> | - | This would contain Client SMTP Ident to be used |
| <ClientEmail><ConnectionTimeout> | - | Value for <ConnectionTimeout> would contain client SMTP server connect timeout value in seconds. Default value is set to 60 seconds. |
| <Language/ AutomaticSpellingCorrection> | - | This would contain true/false value indicating whether to do automatic spelling correction or not. The default value is true. |
| <Language/SpellingTolerance> | - | This would contain true/false value indicating whether to have spelling tolerance or not. The default value is true. |
| <Language/Thresholds/Default> | - | This would contain number indicating spelling tolerance threshold percentage to be used for normal NLP. Valid entries are (0-99) else the default value 15 is used. |
| <Normalization> | - | This would contain normalisation form to be used in V-Engine. Default value is NFC. Possible values are NFC,NFD,NFKC and NFKD. |
| <BOTSTATS> | WriteIntoBOTSTATS  DSNName  UserID  UserPassword  LocalDSNName  LocalUserID  LocalUserPassword  ConvinfoTableName  TransinfoTableName  JourneyinfoTableName  JourneyStepinfoTableName  SurveyTableName  AggregateDataTableName  AggregateAnswerDataTableName  AggregateRecognitionDataTableName  DataBaseType  StatsConvTransTableName  SplitUserInput  CandidateCollectionTableName | This is BOTSTATS node of engine configuration file. Child node indicates WriteIntoBOTSTATS , DSNName, UserID, UserPassword, LocalDSNName, LocalUserID, LocalUserPassword, ConvinfoTableName, TransinfoTableName, JourneyinfoTableName, JourneyStepinfoTableName, SurveyTableName, AggregateDataTableName and AggregateAnswerDataTableName,  AggregateRecognitionDataTableName ,DataBaseType, StatsConvTransTableName, SplitUserInput, CandidateCollectionTableName  of BOTSTATS functionality. |
| < BOTSTATS> < WriteIntoBOTSTATS> | - | If contains “1” enables BOTSTATS writes else if contains “0” disables BOTSTATS writes. |
| < BOTSTATS> < DSNName> |  | This would contain remote data server name as configured in MySQL/MSSQL/Oracle ODBC connector. |
| < BOTSTATS> < UserID> | isSecure | This would contain user id as configured in MySQL/MSSQL/Oracle database. “isSecure” attribute indicates given used id is encrypted or not. Possible values for this attribute are “true” and “false”. |
| < BOTSTATS>< UserPassword> | isSecure | This would contain user password as configured in MySQL/MSSQL/Oracle database. “isSecure” attribute indicates given password is encrypted or not. Possible values for this attribute are “true” and “false”. |
| < BOTSTATS> < LocalDSNName> |  | This would contain local data server name as configured in MySQL/MSSQL/Oracle ODBC connector. |
| < BOTSTATS> < LocalUserID> | isSecure | This would contain user id as configured in MySQL/MSSQL/Oracle database. “isSecure” attribute indicates given used id is encrypted or not. Possible values for this attribute are “true” and “false”. |
| < BOTSTATS>  < LocalUserPassword> | isSecure | This would contain user password as configured in MySQL/MSSQL/Oracle database. “isSecure” attribute indicates given password is encrypted or not. Possible values for this attribute are “true” and “false”. |
| < BOTSTATS>< ConvinfoTableName> | - | Conversation information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| < BOTSTATS>< TransinfoTableName > | - | Transaction information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| < BOTSTATS><DatabaseType> |  | Type of the database used. This field can have either of the two values.   1. MYSQL – For using MYSQL database 2. MSSQL - For Using SQL Server 2005 database 3. ORACLE – For Using Oracle database |
| <BOTSTATS>< JourneyinfoTableName> | - | Journey information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| <BOTSTATS>< JourneyStepinfoTableName  > | - | JourneyStep information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| <BOTSTATS>< SurveyTableName  > | - | Survey information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| <BOTSTATS>< AggregateDataTableName  > | - | AggregateData information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| <BOTSTATS>< AggregateAnswerDataTableName  > | - | AggregateAnswerData information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| <BOTSTATS>< AggregateRecognitionDataTableName |  | AggregateRecognitionData information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| <BOTSTATS>< StatsConvTransTableName> | - | Conversation statistics information table name created in MySQL/MSSQL/Oracle data base for BOTSTATS functionality. |
| <BOTSTATS>< CandidateCollectionTableName > | - | Candidate collection table name created in MySQL/MSSQL/Oracle data base to store collected candidates details for recognitions. |
| <BOTSTATS><  SplitUserInput> | - | If This would contain “true”, logging of split user input is enabled and V-Engine will log split user input in transaction information table. For other values V-Engine will not log split user input in transaction information table. |
| <AcceptExternalSessionID> | - | This would contain values 0/1/2 indicating whether to accept external Session ID or not. Following is the detailed description:-  "2" = Engine accepts and uses externally generated IDENT and continues with NLP.  "1" = Engine ignores externally generated IDENT, generates one itself and continues with NLP. "0" = Engine rejects externally generated IDENT and sends requester to vareject.html  The default value will be “1” |
| <Staging> | - | This would contain true/false value indicating Staging value. The default value is false. |
| <KBValidityThreadSpawnTime> | - | This would contain positive number. Knowledgebase’s Recognitions and Answers date validity is checked for every given value. The default value is 300 seconds. |
| <NumberOfTestSessions> | - | This would contain positive number indicating maximum Test Users count that can access VA Engine at any given time. The default value is 5. |
| <VEngineScriptTimeOut> | - | This would contain positive number indicating V-Engine python script execution time out. |
| <SessionTimeOut> | - | This would contain positive number indicating Session Time Out. The default value is 600 seconds. |
| <ExpiredSessionTimeOut> | - | This would contain positive number indicating for expired session timeout. For how much time expired sessions should be kept in memory. It is in hours. |
| <DefaultTimeoutRecognition> | - | This is the timeout recognition node. This configuration indicates which recognition will be used by engine when it receives session data dependent requests after session timeout. The default value for this configuration is “Timeout”. Possible values for this node are “Timeout” and “Error”. |
| <FAQAlgorithm> | SemanticSuggest  StatisticalSearch  CategorySearch  NLPContinuation | This is the FAQ Algorithm node of the engine configuration file. The child node indicates StatisticalSearch, CategorySearch and NLPContinuation flags for the Related Questions functionality. |
| < FAQAlgorithm><SemanticSuggest> | - | This would contain true/false value indicating whether Related question search using Semantic Suggestis set or not. The default value is false. |
| < FAQAlgorithm>< StatisticalSearch> | - | This would contain true/false value indicating whether Related question search using Statistical Data is set or not. The default value is false. |
| < FAQAlgorithm>< CategorySearch> | - | This would contain true/false value indicating whether Related question search using Category Based Search is set or not. The default value is false. |
| < FAQAlgorithm>< NLPContinuation> | - | This would contain true/false value indicating whether Related question search using NLP search continuation is set or not. The default value is false. |
| <CategoryMaxNoOfFAQ> | - | This would contain a positive number indicating the maximum number of Category FAQS to be fetched. |
| <SemanticMaxNoOfFAQ> | - | This would contain a positive number indicating the maximum number of Semantic FAQS to be fetched. |
| <StatisticalMaxNoOfFAQ> | - | This would contain a positive number indicating the maximum number of Statistical FAQS to be fetched. |
| <NLPContMaxNoOfFAQ> | - | This would contain a positive number indicating the maximum number of NLP Continuation FAQS to be fetched. |
| <MaxConditionRankForFAQ> | - | This would contain a positive number indicating the maximum condition rank for the FAQs to be fetched |
| <StatisticalData> | Bandwidth DevaluationFactor ReverseDevaluationFactor SelfReferencingDevalFactor FreeTextPromotionFactor IncludeNRinDevaluationSet ClickedBasicWeight MinimumWeight StatisticalInfoTableName | This is the Statistical data node of the engine configuration file. Child Node indicates Bandwidth, DevaluationFactor, ReverseDevaluationFactor, SelfReferencingDevalFactor, FreeTextPromotionFactor, IncludeNRinDevaluationSet, ClickedBasicWeight, MinimumWeight, StatisticalInfoTableName |
| <StatisticalData> <Bandwidth> | - | This would contain a positive number indicating the bandwidth for statistical logging. |
| <StatisticalData> <DevaluationFactor> | - | This would contain a positive number indicating the Devaluation Factor for statistical logging. |
| <StatisticalData> <ReverseDevaluationFactor> | - | This would contain a positive number indicating the Reverse Devaluation Factor for statistical logging. |
| <StatisticalData> <SelfReferencingDevalFactor> | - | This would contain a positive number indicating the Self Referencing Devaluation Factor for statistical logging. |
| <StatisticalData> <FreeTextPromotionFactor> | - | This would contain a positive number indicating the Free Text Promotion Factor for statistical logging. |
| <StatisticalData> <IncludeNRinDevaluationSet> | - | This would contain a positive number indicating Include non-recommendable Q/A pairs into devaluation set for statistical logging. |
| <StatisticalData> <ClickedBasicWeight> | - | This would contain a positive number indicating Clicked Basic Weight for statistical logging. |
| <StatisticalData> <MinimumWeight> | - | This would contain a positive number indicating Minimum Weight for statistical logging. |
| <StatisticalData> <StatisticalInfoTableName> | - | Statistical Info Table Name created in MySQL/MSSQL data base |
| <PipeThresholdValue> | - | This would contain a positive number indicating the threshold value for pipe operator. |
| <IsTestEngine> | - | This would contain true/false value indicating whether it is a test engine. The default value is false. |
| <MaxNoOfRecognition> | - | This would contain a positive value indicating the maximum number of recognitions for index based search |
| <DynamicAnswerTableName> | - | Dynamic Answer Table Name created in MySQL/MSSQL data base |
| <ActivityLogTableName> | - | Activity Log Table Name created in MySQL/MSSQL data base |
| <DefaultAdditionalConditionForFAQ> | - | This would contain true/false value indicating the default test result value for additional conditions encountered when finding related questions. |
| <CacheSemanticAnswerResult> | - | This would contain true/false value indicating whether to cache semantic answer result for semantic search or not. |
| <MacroIndexing> | - | This is the MacroIndexing Node of the engine configuration file. Child Node indicates UseMacroIndexing, MacroIndexingForAlternates, MacroIndexingForAnswers, MacroIndexingForMacros, MacroIndexWordWeight, RealTimeMode, BatchMode, IgnoreListFilePath and TableNames |
| <MacroIndexing> <UseMacroIndexing > | - | This would contain true/false value indicating whether Macro Indexing is ON or OFF. The default value is false. |
| <MacroIndexing> <MacroIndexingForAlternates> | - | This would contain true/false value indicating whether Macro Indexing for alternates is ON or OFF. The default value is false. |
| <MacroIndexing> <MacroIndexingForAnswers> | - | This would contain true/false value indicating whether Macro Indexing for answers is ON or OFF. The default value is false. |
| <MacroIndexing> <MacroIndexingForMacros> | - | This would contain true/false value indicating whether Macro Indexing for macros is ON or OFF. The default value is false. |
| <MacroIndexing> <MacroIndexWordWeight> | - | This would contain a positive value indicating the Word Weight to be used for Macro Indexing. |
| <MacroIndexing> <RealTimeMode> | - | This would contain true/false value indicating whether Real Time Macro Indexing is ON or OFF. The default value is false. |
| <MacroIndexing> <BatchMode> | - | This would contain true/false value indicating whether Batch Mode Macro Indexing is ON or OFF. The default value is false. |
| <MacroIndexing> <IgnoreListFilePath> | - | This would contain the Ignore List file path fo0r Macro Indexing.  E.g.  ../kb/ignore\_list.txt |
| <MacroIndexing> <TableNames> | - | This is the Table Name node of Macro Indexing. Child node indicates WordDet, MacroDet, AnsWordDet, AnsMacroDet, AlternateWordMap, AlternateMacroMap, AnswerWordMap and AnswerMacroMap. |
| <MacroIndexing> <TableNames> <MacroData> | - | Macro Data table name created in MySQL database for Macro Indexing functionality to store Macro Id and Macro Name. |
| <MacroIndexing> <TableNames> <WordData> | - | Word Data table name created in MySQL database for Macro Indexing functionality to store word Id and word Name. |
| <MacroIndexing> <TableNames> <AnswerHash> | - | Answer Hash table name created in MySQL database for Macro Indexing functionality to store Answer Id and Answer plain text hash value. |
| <MacroIndexing> <TableNames> <AlternateHash> | - | Alternate Hash table name created in MySQL database for Macro Indexing functionality to store Alternate Id and Alternate text hash value. |
| <MacroIndexing> <TableNames> <MacroHash> |  | Macro Hash table name created in MySQL database for Macro Indexing functionality to store Macro Id and Macro condition string hash value. |
| <MacroIndexing> <TableNames> <AlternateWordIndex> | - | Alternate Word Index table created in MySQL database for Macro Indexing functionality to store Alternate word index data. |
| <MacroIndexing> <TableNames> <AlternateMacroIndex> | - | Alternate Macro Index table created in MySQL database for Macro Indexing functionality to store Alternate macro index data. |
| <MacroIndexing> <TableNames>  < AnswerWordIndex> | - | Answer Word Index table created in MySQL database for Macro Indexing functionality to store Answer word index data. |
| <MacroIndexing> <TableNames>  < AnswerMacroIndex> | - | Answer Macro Index table created in MySQL database for Macro Indexing functionality to store Answer macro index data. |
| <MacroIndexing> <TableNames>  < MacroIndexStatus> | - | Macro index status table created in MySQL database for Macro indexing functionality to store macro index data status. Like. To store Engine MI data write in progress status, V-Portal MI data read in progress status |
| <MacroIndexing> <DBLogRetryInterval> | - | This will contain value of interval between macro index data insert retries. Engine will wait and retry after configured interval if V-Portal read in progress for macro index data. |
| <MacroIndexing>  <DBInsertRowsAtOnce> | - | This will contain value of number of rows to insert in macro indexing DB at once to improve record insertion performance. |
| <AutoComplete> | - | This is the Autocomplete Node of the engine configuration file. Child Node indicates TableNames |
| <AutoComplete> <TableNames> | - | This is the Table Name node of Macro Indexing. Child node indicates AutoCompleteAlternateWordIndex, AutoCompleteAnswerQuestionIndex, AutoCompleteRecognitionWordIndex, AutoCompleteRecognitionAlternateIndex,AutoCompleteAlternateHash, AutoCompleteRecognitionHash, AutoCompleteAnswerQuestionHash, AutoCompleteWordData and AutoCompleteStatus. |
| <AutoComplete> <TableNames>  <AutoCompleteAlternateWordIndex> |  | Condition Alternate Word Index table created in database for autocomplete functionality to store Alternate word index data. |
| <AutoComplete> <TableNames>  <AutoCompleteAnswerQuestionIndex> | - | Answer question Word Index table created in database for autocomplete functionality to store Answer question word index data. |
| <AutoComplete> <TableNames>  <AutoCompleteRecognitionWordIndex> | - | Recognition question Word Index table created in database for autocomplete functionality to store Recognition question word index data. |
| <AutoComplete> <TableNames>  <AutoCompleteRecognitionAlternateIndex> | - | Recognition Alternate Word Index table created in database for autocomplete functionality to store Alternate word index data. |
| <AutoComplete> <TableNames>  <AutoCompleteAlternateHash> | - | Alternate Hash table name created in database for autocomplete functionality to store Alternate Id and Alternate text hash value. |
| <AutoComplete> <TableNames>  <AutoCompleteRecognitionHash> | - | Recognition question Hash table name created in database for autocomplete functionality to store Recognition Id and Recognition question text hash value. |
| <AutoComplete> <TableNames>  <AutoCompleteAnswerQuestionHash> | - | Answer question Hash table name created in database for autocomplete functionality to store Answer Id and Answer question text hash value. |
| <AutoComplete> <TableNames>  <AutoCompleteWordData> | - | Word Data table name created in database for autocomplete functionality to store word Id and word Name. |
| <AutoComplete> <TableNames>  <AutoCompleteStatus> | - | Autocomplete index status table created in database for autocomplete functionality to store autocomplete index data status. Like. To store Engine autocomplete data write in progress status, V-Portal autocomplete data read in progress status |
| <AutoComplete> <ResponseFormat> | - | This would contain autocomplete response format to be returned for autocomplete request. Supported values are 'xml' and 'json'. |
| <ModuleID> | - | This would contain Module Id of V-Engine instance |
| <EngineID> | - | This would contain unique V-Engine ID string for that V-Engine instance. This is useful when multiple V-Engines are configured to write in same remote Database. |
| <DumpDictionary> | - | If This would contain “true”, logging/dump of dictionary words is enabled and V-Engine will log/dump dictionary words in file after Knowledgebase load. Dictionary dump file will be created at configured transcript file path. For other values V-Engine will not log/dump dictionary words. |
| <DeepMatcherLogging> |  | If This would contain “true”, logging of deep matcher functionality is enabled, and V-Engine will log details in file for deep matcher function calls. Log file will be created at configured log directory. |
| <Vbuilder> | - |  |
| <SystemScriptVariables> | - | This will contain the value “true” or “false”. This value can be used within the VA Engine by accessing the system variable “Vbuilder”. Default value for this field is “false”. |
| <SystemScriptVariables/Variable> | - | This node will contain the system variables which need to be loaded when the VA Engine is started. |
| <SystemScriptVariables/Variable/Name> | - | This is the parent node which will store the System Variable name and value nodes. |
| <SystemScriptVariables/Variable/Value> | - | This node will contain the system variable name. |
| <LicenseExpAlerts> | - | This node will contain license expiry alter configurations. |
| <LicenseExpAlerts><DaysForLicenseExpAlert> |  | Days for alert before License Expiry. |
| <LicenseExpAlerts><AlertEmailIDs> |  | E-mail Ids to which License Expiry alert E-mail to be sent. |
| <LicenseExpAlerts><LicenseContact> |  | Contact E-mail for license expiry |
| <EncryptedRequestResponse> |  | This would contain true/false value. If “true”, engine will send encrypted response, expecting incoming encrypted user requests. Default value is false. This configuration is applicable only to non-test engines. |

Table 2:Nodes of “VirtualAssistant-VEngine\_Config.xml”

Sample “*VirtualAssistant-VEngine\_Config.xml*” is given below:



Following table describes important nodes of “*language.xml*” configuration file:

|  |  |  |
| --- | --- | --- |
| Node Name | Attribute Name | Description |
| <language\_configuration> | - | This is main node of language configuration file name. |
| <general/language> | value | The value attribute would contain the language name. |
| <general/language> | locale\_language | This is an optional attribute. locale\_language attribute would contain the lower-case language codes as defined by ISO-639. In case this attribute is not specified the locale language of the system will be taken by default.  Example value for Turkish language is tr |
| <general/language> | locale\_country | This is optional attribute. locale\_country attribute would contain the upper-case country codes as defined by ISO-3166. In case this attribute is not specified the locale country of the system will be taken by default.  Example value for Turkey is TR  Please check the following links for the codes:  <http://www.iso.org/iso/country_codes>  <https://www.iso.org/obp/ui/#search> |
| <general /sentence\_delimiters> | value | The value attribute would contain the sentence delimiters. |
| <general /word\_delimiters> | value | The value attribute would contain the word delimiters. |
| <general /keep\_delimiters> | value | The value attribute would contain the keep word delimiters. This is added to remove hard coded list in V-Engine for word delimiter handling. |
| <general /nonsense\_characters> | value | The value attribute would contain the nonsense characters. |
| <general /nonsense\_min\_length> | value | The value attribute would contain the nonsense minimum length. |
| <general/ignore\_characters> | value | The value attribute would contain characters to be ignored when case insensitive comparison done for user input against autocomplete candidate text during autocomplete process. |
| <general/decimal\_sep> | value | The value attribute would contain the string for decimal separator. |
| <general/thousands\_sep> | value | The value attribute would contain the string for thousands separator. |
| <key\_mapping> | - | This would contain the list of entry node. It contains keyboard mapping for given language. It used for auto correcting a word. |
| <character\_normalisation> | - | This would contain character normalization list. |
| <word\_part\_distances> | - | This would contain word part distances list. |
| <splitting> | value | This would contain either ‘true’ or ‘false’.  This turns the entire splitting procedure on or off. If ‘false’, only the normal separation character based sentence and word splitting takes place. |
| <splitting/separation\_mapping> | - | This would contain list of mappings from unseparated words to separation results (with spaces in between).  The separation process always looks here first. If found, the separation result is taken as it is without further splitting (regardless of the words being known or not) |
| <splitting/separated\_word\_exclusion> | - | This would contain list of words to be excluded.  Words in this list can never turn up in a separation result from prefix-, suffix- or dictionary-splitting. The only exception is when it explicitly occurs in the separation mapping list, which has precedence. |
| < splitting /autocorrect\_before\_separation> | value | This would contain either ‘true’ or ‘false’.  With this setting ‘true’, if a word is not found in the dictionary (after simplification), an auto-correction attempt is made before a prefix, suffix or dictionary based separation attempt is made and if a match is found splitting is skipped entirely.  With the setting ‘false’, if a word is not found in the dictionary (after simplification), a separation attempt is made. Still auto-correction of the unsplit word is tried as a part of the separation attempt. |
| < splitting /tolerance\_before\_separation> | value | This would contain either ‘true’ or ‘false’.  With this setting ‘true’, if a word is not found in the dictionary (after simplification), an attempt is made to find the word with spelling tolerance before a prefix, suffix or dictionary based separation attempt is made and if a match is found splitting is skipped entirely.  With the setting ‘false’, if a word is not found in the dictionary (after simplification), a separation attempt is made. Still spelling tolerance on the unsplit word is tried as a part of the separation attempt. |
| <splitting/single\_character\_handling> | value | This would contain either ‘true’ or ‘false’.  Switches on/off the feature of special handling for languages that don’t know word boundaries (Chinese, Japanese for example). Parts of the input belonging to these languages will be split into individual characters. |
| <single\_character\_handling /unicode\_ranges> | - | This would contain list of hexadecimal integer ranges.  Characters in these ranges will be split off individually if single character special handling is on. |
| <splitting/allow\_unknown\_words> | value | This would contain either ‘true’ or ‘false’.  If this is ‘true’, all parts that a word is split into (apart from single characters dealt with by the special handling) must be known one way or the other (dictionary, dictionary plus spelling tolerance, prefix-, suffix-list).  If it is ‘false’, the splitting result can contain unknown parts (but for each splitting into two parts at least one of the parts needs to be known). |
| <splitting  /minimum\_separated\_unknown\_length> | value | This would contain positive integer.  Only applies if ‘allow unknown word in splitting result’-option is on. If an unknown part of the splitting result would be smaller than the configured minimum length, the splitting will not take place.  Does not apply to single character special handling.  It makes no sense to set this to a value smaller than the ‘minimum separated word length’ setting, as that will take precedence. |
| <splitting  /minimum\_separated\_known\_length> | value | This would contain positive integer.  If a part of the splitting result would be smaller than the configured minimum length, the splitting will not take place.  It is not recommended to set this to 1, as usually all letters of the alphabet turn up in the normal dictionary.  Does not apply to single character special handling. It does applies to prefix and suffix splitting as well, but not to the prefixes and suffixes themselves. |
| <splitting/prefix\_splitting> | value | This would contain either ‘true’ or ‘false’.  If ‘true’, the separation process tries to split off known prefixes (from the prefix list) from the beginning of the word. |
| <prefix\_splitting/max\_prefixes> | value | Words may have multiple (chained) prefixes. This value defines how many prefix splitting attempts the procedure tries. ‘0’ means no limit. |
| <prefix\_splitting/prefix\_list> | - | This would contain list of words.  List of prefix words that can be split off. |
| <splitting/suffix\_splitting > | value | This would contain either ‘true’ or ‘false’.  If ‘true’, the separation process tries to split off known suffixes (from the suffix list) from the end of the word. |
| <suffix\_splitting/max\_suffixes> | value | Words may have multiple (chained) suffixes. This value defines how many suffix splitting attempts the procedure tries. ‘0’ means no limit. |
| <suffix\_splitting/suffix\_list> | - | This would contain list of words.  List of suffix words that can be split off. |
| <splitting/word\_list> | - | This would contain list of words.  Words in this list will be added to the internal dictionary that otherwise is built out of all word conditions. These added words can be used in splitting, but also for spelling tolerance matching. |
| <splitting/autocorrect\_on\_parts> | value | This would contain either ‘true’ or ‘false’.  If this feature is on and the procedure has split off one known part, it allows autocorrect matches on the second part. |
| <splitting/tolerance\_on\_parts> | value | This would contain either ‘true’ or ‘false’.  If this feature is on and the procedure has split off one known part, it allows matches with spelling tolerance on the second part. |
| <splitting/headword\_position> | value | This would contain ‘start’ or ‘end’ or ‘undefined’.  For headword position ‘start’ headword is the first splitting part that isn't a prefix.  For headword position 'end' it's just the same, only with the headword being the last splitting part that isn't a suffix.  For headword position 'undefined' only the known score is used for comparison. |
| <splitting/  part\_type\_value\_autocorrection> | value | This would contain real number, 0 <= x <= 1. |
| <splitting/  part\_type\_value\_spelling\_tolerance> | value | This would contain real number, 0 <= x <= 1. |

Table 3: Nodes of “language.xml”

Sample “*language.xml*” is given below:



# RUNNING the java web-interface server

The Java Web Interface server – Tomcat - needs to be up and running before executing the web application.

## Setup UILayer (cv.war) in Tomcat

### Configuration Setup

Perform the following steps below to setup the configuration required by UILayer (cv.war) application. Please note, the cv.war should be deployed in the Tomcat and should be exploded by Starting and Stopping Tomcat Server once.

1. The web configuration files that need to be configured for the application include:
2. ..\cv\config\Web\_Config.xml
3. ..\cv\web.xml
4. ..\cv\config\log4j.xml
5. ..\cv\config\VAProperties.properties
6. ..\ apache-tomcat-6.0.18\conf\tomcat-users.xml
7. ..\ apache-tomcat-6.0.18\conf\server.xml
8. ..\cv\js\cvutil.js
9. Edit the contents of **Web\_Config.xml** as per the description given below.

|  |  |  |
| --- | --- | --- |
| Node Name | Attribute Name | Description |
| <Web\_Config/Domain> | - | This would contain the domain name of the server.  E.g. mastek.com |
| <ConnectionInfo/NameServiceAddress> | - | This would contain the address of the Naming Service for connection with the engine factory.  E.g. corbaloc::172.16.232.172:9990/NameService |
| <ConnectionInfo/EngineFactoryBindName> | - | This would contain the path binding the engine factory to the name service.  E.g. CreativeVirtual/EngineFactory |
| <URLs/URL> | name | This would contain the name of the URL to be displayed to the user.  E.g. CLOSED\_URL |
| path | This would contain the path of the web page corresponding to the mentioned URL name present on the tomcat server.  E.g. vaclosed.html |
| <EngineFailure/Email> | id | This would contain the email id of the recipient  E.g. tom@creativevirtual.com |
| name | This would contain the name of the recipient  E.g. Tom |
| <SMTPInfo/SMTPServer> | - | This would contain the IP address of the SMPT server for sending emails  E.g. 172.16.232.172 |
| <SMTPInfo/SMTPPort> | - | This would contain the port number of the SMTP server  E.g. 25 |
| <SMTPInfo/SMTPFrom> | - | This would contain the email sender’s name  E.g. nick@creativevirtual.com |
| <Default/Allow> | - | This would contain the list of IP Addresses which would be granted access to the application  E.g. 127.0.0.1, 172.16.232.179 |
| <Default/Deny> | - | This would contain the list of IP Addresses which would be denied access to the application  E.g. 127.0.0.1, 172.16.232.171/178 |
| <IPRestrictions/ReqMode> | type | This would contain the user’s request mode.  E.g. normal, admin, trigger, state |
| <ReqMode/Allow> | - | This would contain the list of IP Addresses which would be granted access to the application for the specific request mode type specified in the ‘type’ attribute  E.g. 127.0.0.1, 172.16.232.181 |
| <ReqMode/Deny> | - | This would contain the list of IP Addresses which would be denied access to the application for the specific request mode type specified in the ‘type’ attribute  E.g. 127.0.0.1, 172.16.232.180 |
|  |  |  |

Table 4: Nodes of “Web\_Config.xml”

Sample “*Web\_Config.xml”* is given below:



1. Edit the contents of web.xml as per the description given below.

|  |  |  |
| --- | --- | --- |
| Node Name | Attribute Name | Description |
| <context-param/param-name /param-value> | - | This would contain the module id which would be the same as the module id of the V-Engine that the web application would be connecting with, present in the EngineFactory.xml.  E.g.  <context-param>  <param-name>  ClientDetails  </param-name>  <param-value>  **CreativeVirtual**  </param-value>  </context-param>  **Note:** The <param-value> **CreativeVirtual**  </param-value>  needs to be changed in both the existing instances. |
| <security-role\role-name> | - | Role Names authorized to access the web application.  Eg.  <security-role>  <role-name>admin</role-name>  </security-role> |
| <security-constraint/auth-constraint/role-name> | - | Role Name authorized to access the administrator application. This role-name should be present in tomcat-users.xml [Ref 5]  Eg.  <auth-constraint>  <role-name>admin</role-name>  </auth-constraint> |
| <welcome-file-list\welcome-file > |  | This would contain the file name which will load when we open UILayer. This needs to be set in the following manner for non sso and sso setup respectively:  ***For Non SSO UILayer***  <welcome-file-list>  <welcome-file>index.jsp</welcome-file>  </welcome-file-list>  ***For SSO UILayer***  <welcome-file-list>  <welcome-file>welcome.jsp</welcome-file>  </welcome-file-list> |
| <context-param/param-name/ param-value> | contextConfigLocation | Configuration file name which will be used to run UILayer under Non SSO/SSO modes.  ***Eg.1 For Non SSO UILayer***  /WEB-INF/security-config.xml  ***Eg.2 For SSO UILayer***  /WEB-INF/security-config-sso.xml |

Table 5: Nodes of “web.xml”

Also please note for caching related support, please check and follow the instructions mentioned in the attached web.xml file.

Sample “w*eb.xml”* is given below:



1. Edit the log4j.xml file present in config directory of the application to change the name of the log file to make it client specific.

E.g. ../UILayer/config/log4j.xml

To change the name of the log file generated to BtDiagnostic, change the file name in log4j.xml as given

<appender name="VA\_ROLLING\_FILE" class="org.apache.log4j.DailyRollingFileAppender">

<param name="File" value="***../logs/BtDiagnostic.log***" />

1. Edit the VAProperties.properties present in config directory of the application to set the server type and other properties as follows:

|  |  |
| --- | --- |
| Property Name | Property Description and Sample Value |
| *server.type* | This property will used to specify on which web server UILayer will run. Comment out the entry “server.type=IBM”  Sample Value:  *server.type=* *TOMCAT* |
| *sso.failure.redirection* | This property will be used to force SAML based authentication for the end user. **Set it to ON only if deployed in SSO environment system.**  Sample Value: OFF |
| *auth.type* | Authorization type, any of "OWN", "SAML 2.0", "OAUTH 2.0". Sample Value: SAML 2.0 |
| *auth.saml.tokenValid* | This URL validates the current token is valid or not Sample Value: http://openam.creativevirtual.com:8080/openam/identity/isTokenValid?tokenid= |
| *auth.saml.login* | This is SAML login URL to start SSO Sample Value: <http://uilayer.creativevirtual.com/cv/saml/login> |
| *auth.saml.sp.spName* | This indicates the service provide name configured in OpenAM Sample Value: sp\_uilayer |
| *auth.saml.logout* | This indicates the global Logout URL Sample Value: http://openam.creativevirtual.com:8080/openam/identity/logout?subjectid= |
| *auth.saml.discovery* | This URL is helpful in discovering the IDP configured in OpenAM  Sample Value: <http://uilayer.creativevirtual.com/cv/saml/discovery/alias/myapp> |
| *auth.saml.metadata* | This is to get Meta-Data of application metadata Sample Value: <http://openam.creativevirtual.com:8080/openam/saml2/jsp/exportmetadata.jsp?entityid=http://openam.creativevirtual.com:8080/openam> |
| *auth.saml.alias* | This is required by Discovering IDPs Sample Value: <http://uilayer.creativevirtual.com/cv/saml/login/alias/myapp?disco=true> |
| *validate.ssotoken.onclick* | This property is used to validate SSO token on every request, set it to Y to enable this feature.  Eg.  *validate.ssotoken.onclick=N* |
| *sso.cookie.name* | Name of the cookie which will be set by the IdP. If this cookie is detected, system tries to authenticate user by SSO by default. Sample Value: iPlanetDirectoryPro |
| *sso.user.attribute.fname* | Name of the SAML assertion attribute in which end user First Name will be sent by the IdP.  Sample Value: givenname |
| *sso.user.attribute.lname* | Name of the SAML assertion attribute in which end user Last Name will be sent by the IdP. Sample Value: sn |
| *sso.user.attribute.email* | Name of the SAML assertion attribute in which end user email will be sent by the IdP. Sample Value: mail |
| *sso.value.nameid* | This property will be used when VA is under SSO and it should be set to the value which will be returned as NAMEID value in SAML response.  Eg,  *sso.value.nameid=username* |
| *engine.request.encrypt* | This property will specify engine requests will be encrypt or not.  Sample Value:  *engine.request.encrypt=OFF* |

1. Add the following contents in tomcat-users.xml to configure users to access the Administrator application.

Eg.

<tomcat-users>

<role rolename="admin"/>

<user username="admin" password="password" roles="admin"/>

</tomcat-users>

|  |  |  |
| --- | --- | --- |
| Node Name | Attribute Name | Description |
| <role> | rolename | This would be the name of the role which would be authorised to access the administrator application |
| <user> | username | This would be the username of the user intending to access the administrator application |
| password | This would be the password of the user accessing the administrator application |
| roles | This would be the role name for the user |

Table 6: Nodes of “tomcat-users.xml”

Mulitple roles and multiple users can be configured as shown below:

Eg.

<role rolename="manager"/>

<role rolename="admin"/>

<user username="manager" password="manager" roles="manager"/>

<user username="admin" password="admin" roles="admin"/>

Sample “*tomcat-users.xml”* is given below:



**NOTE:**

To add the password in an encrypted format in tomcat-users.xml, do as given below

1. Run the digest batch file present in bin directory of the tomcat installation directory

Eg. D:\apache-tomcat-6.0.18\bin\digest.bat

* + Open the command prompt and enter the directory path of the digest.bat file

Eg. cd D:\apache-tomcat-6.0.18\bin

* + Type the command ***digest –a*** (algorithm) ***algorithm name*** ***password value***

Eg. digest –a sha password

***sha*** – Name of the algorithm

***password value*** – password

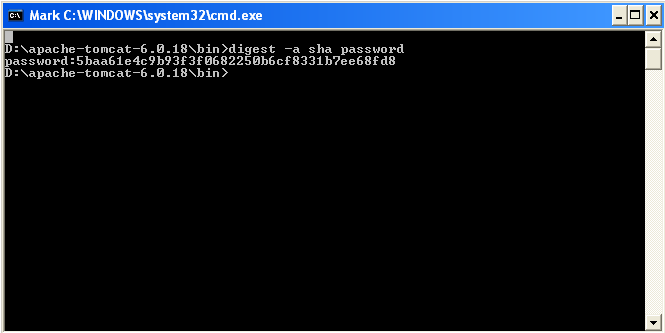


Figure 14 : Password Generation

1. The encrypted password value generated as shown in the Figure 1 can then be added in the password attribute of tomcat-users.xml.
2. Edit the contents of server.xml as given below only if the password being added in tomcat-users.xml is in an encrypted format.

For the Realm with className and resource name as given below

Eg.

<Realm className="org.apache.catalina.realm.UserDatabaseRealm"

resourceName="UserDatabase"/>

Add the name of the algorithm that you would be using to encrypt the passwords.

Eg.

<Realm className="org.apache.catalina.realm.UserDatabaseRealm"

***digest="sha"***

resourceName="UserDatabase"/>

1. Update the contents of **cvutil.js** as given below:

|  |  |
| --- | --- |
| Node Name | Description |
| var isLiveChatEnabled=false | This variable is used to configure if Live chat Handover is required.  Set **true** if LiveChat handover is required. Set false if LiveChat handover is not required.  Please note, appropriate knowledgebase will also need to be configured for the livechat handover. |
| var initiateLcEndUserSSO=false | This variable is used to configure whether the live chat end user is under SSO or not.  If it is set to true then at the time of LiveChat handover, it will open popup to check SSO user credentials. |
| var livechatUrl | This would be variable which contain Live chat application (LCUIServer) URL for LC Handover.  Eg.  var livechatUrl = "http://livechat.creativevirtual.com/livechat/"; |
| var websocketUrl | This would be variable which contain Live chat application (ChatServer) URL for LC Handover.  Eg.  var livechatUrl = "http://livechat.creativevirtual.com/ChatServer /"; |
| var va\_url | This would be variable which contain UILayer application (LCUIServer) URL for LC Handover.  Eg.  var livechatUrl = "http://livechat.creativevirtual.com/livechat/"; |
| var initiateVAUnderSSO | Set this variable to true if VA is going to run under SSO, Otherwise set it to false.  Eg.  var initiateVAUnderSSO = false; |

1. Start the tomcat server.

**Note: Please make sure that the Naming Service, Engine Factory and the V-Engine are up and running before continuing with the next set of instructions.**

## Run the Web Application

To run the deployed web application, perform the following

1. Open a new browser and type the following URL in the address bar

http://<HOSTNAME>:<PORTNUMBER>/<context-root>/index.jsp to see the screen as shown below

**HOSTNAME:** Hostname of the machine

**PORTNUMBER:** HTTP Port Number

**context-root:** Name of the deployed war file

E.g**.** The url to access the application when cv.war has been deployed should be as given

[**http://localhost:8080/cv/index.jsp**](http://localhost:8080/cv/index.jsp)

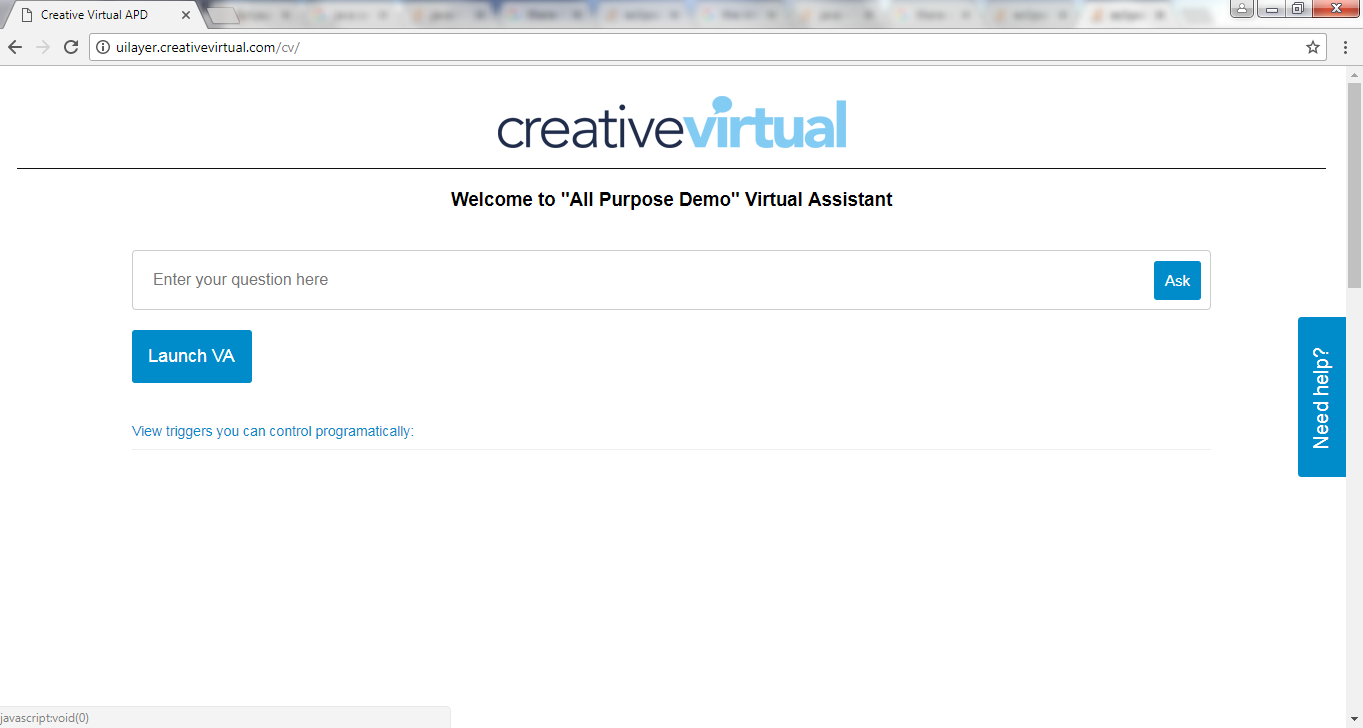


Figure 15: Start VA

1. Click on Need Help to view VA

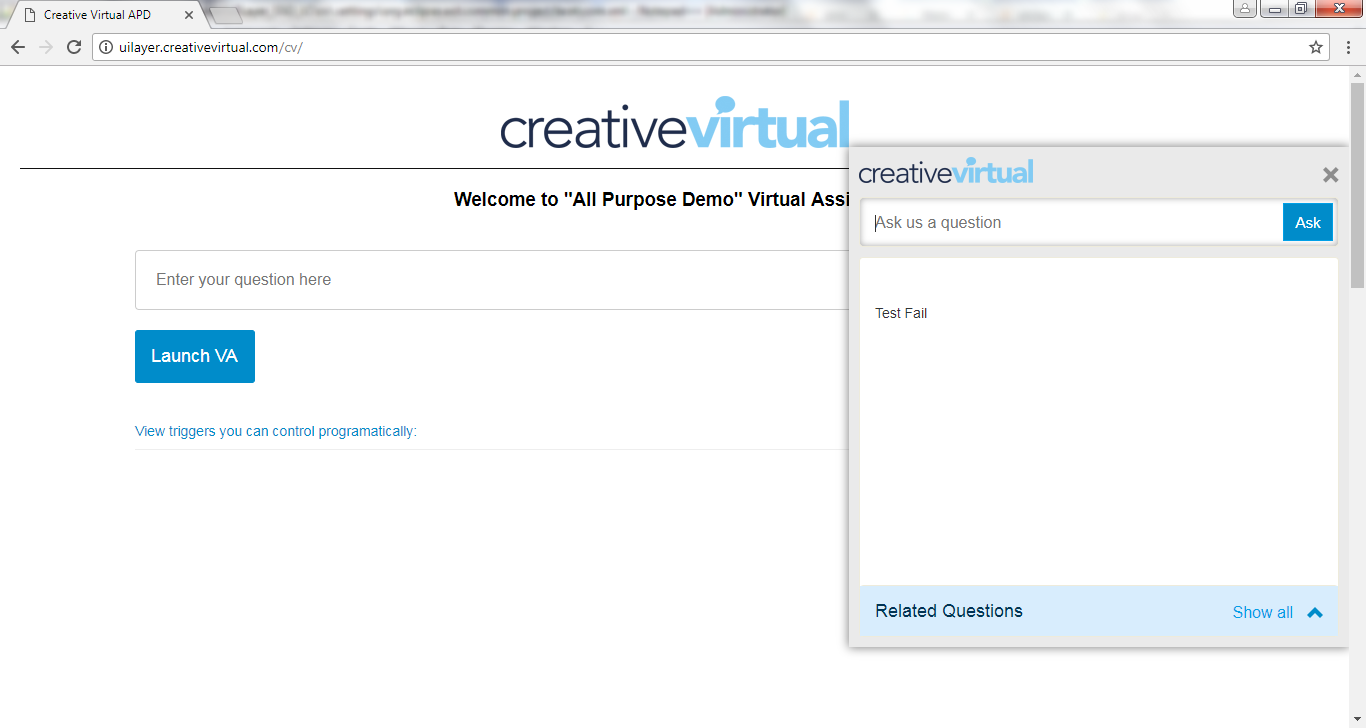


Figure 16: Chat with VA Avatar

## Run the Administrator Application

To run the deployed web application, perform the following

Note: Please make sure that the Naming Service, Engine Factory and the V-Engine are up and running before continuing with the next set of instructions.

1. Open a new browser and type the following URL in the address bar

http://<HOSTNAME>:<PORTNUMBER>/<context-root>/admin.htm to see the screen as shown below

**HOSTNAME:** Hostname of the machine

**PORTNUMBER:** HTTP Port Number

**context-root:** Name of the deployed war file

E.g**.** The url to access the application when cv.war has been deployed should be as given

[**http://localhost:8080/cv/admin.htm**](http://localhost:8080/cv/admin.htm)

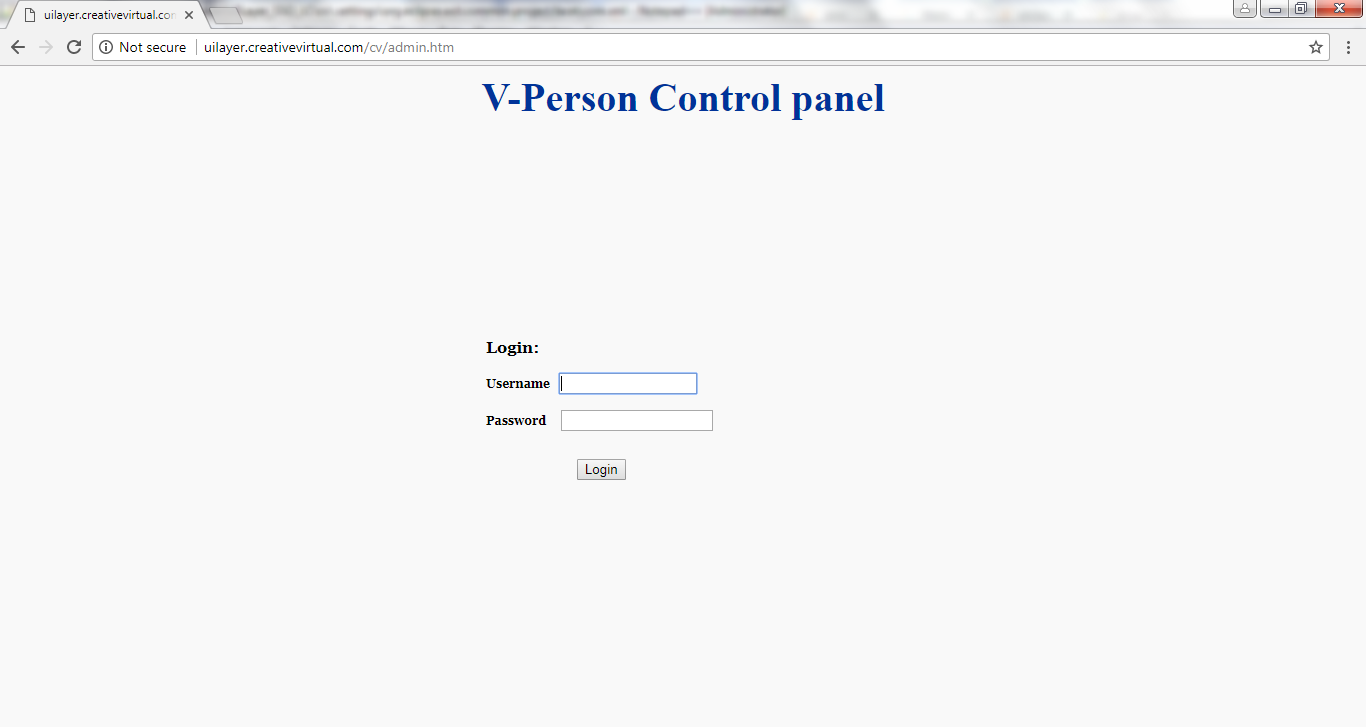


Figure 17: Administrator Login Screen

1. Enter the username and password configured in “*tomcat-users.xml”* and click on the login button. For encrypted password configured, add the plain text password.

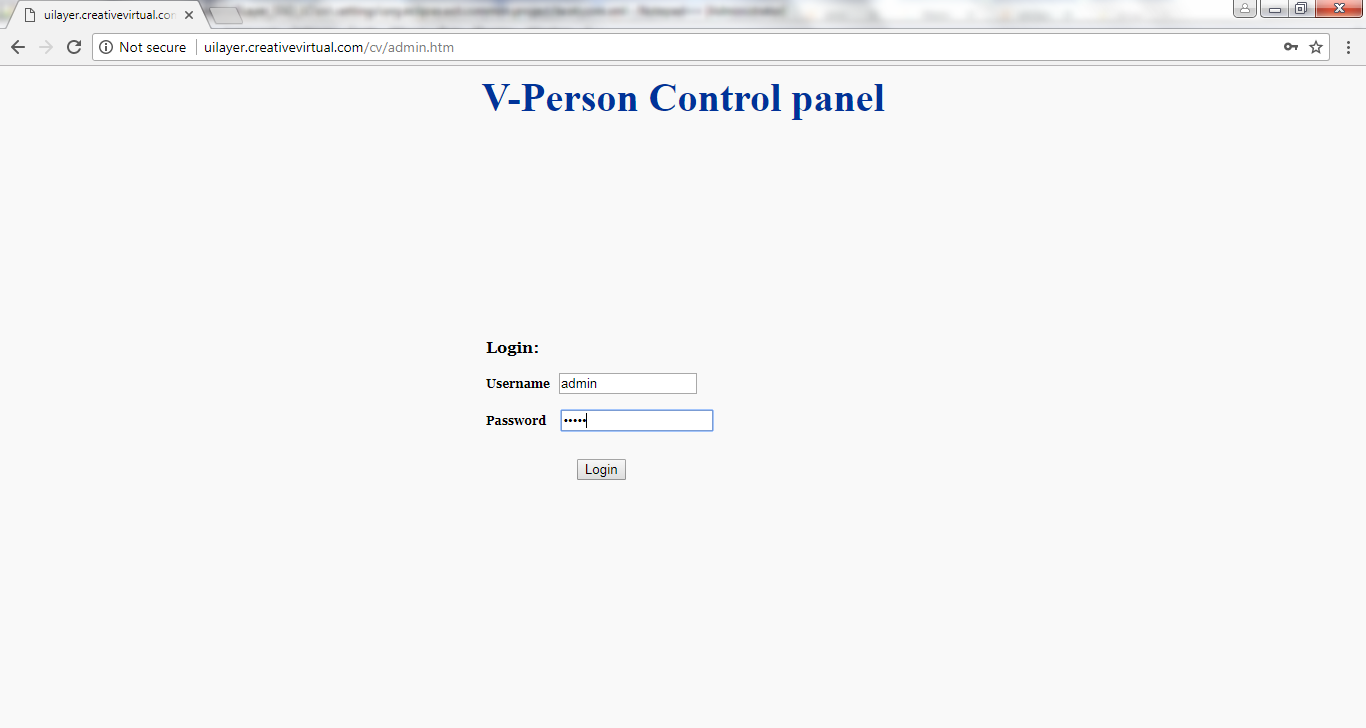


Figure 18: Authentication Details

1. The Administrator Screen is displayed on successful authentication as shown below

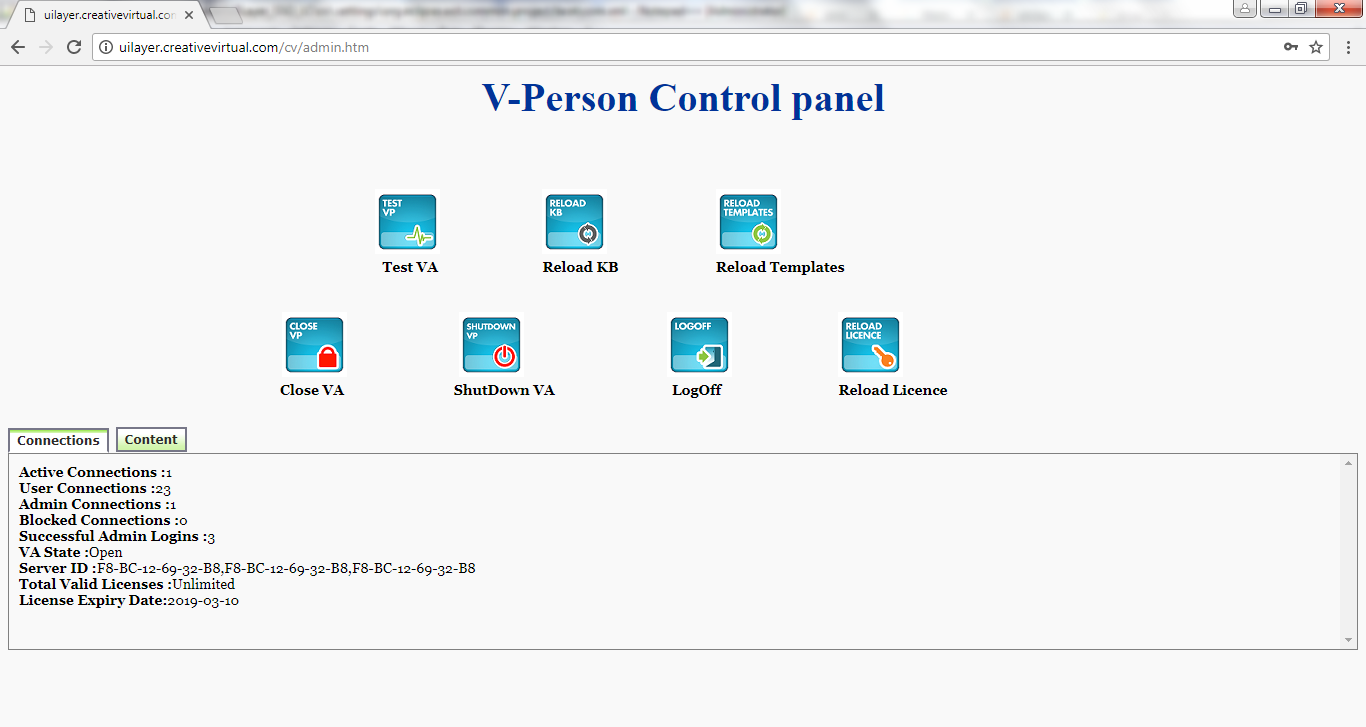


Figure 19: Administrator Details

## Setup for SSO for UILayer using OpenAM

Please ignore the steps in this section in case if we are going to deploy NON-SSO UILayer.

Follow the steps mentioned in the below embedded presentation to setup SSO (OpenAM) on another tomcat instance. Please note, the following document may contain some references to livechat installation, you are required to please replace livechat with uilayer.



## Tomcat Security Issues & Resolutions (Optional)

**Note: The following steps needs to be performed only when the security issues of tomcat server are to be addressed.**

* **Default HTTP Server Error messages are displayed**

EXPLANATION:

Application displays default HTTP server error messages. These messages contain server details.

To mitigate this risk, follow the steps given

* + 1. For all existing applications already deployed on the tomcat server,

E.g. D:\apache-tomcat-6.0.18\webappsmanager,host-manager,ROOT etc

modify the web.xml for each of the deployed applications to add the following before the </web-apps> tag.

<error-page>

<error-code>404</error-code>

<location>/file\_not\_found.html</location>

</error-page>

* + 1. Add a custom file\_not\_found.html into each of these applications.

E.g. webapps/manager/file\_not\_found.html

* + 1. In the existing manager application present in webapps directory in the tomcat installation, update the 401.jsp present, with custom messages.
* **Default server administrative page is available**

EXPLANATION:

The default web server administrative page is available to all users.

To mitigate this risk, follow the steps given

1. Update the index.html present in the ROOT application already deployed on the tomcat server with a custom page.
2. Delete the index.jsp present in the application.

* **Sample applications are deployed and running on the web server**

EXPLANATION:

The web server has sample applications deployed and running. An attacker can exploit vulnerabilities in these sample applications to attack the underlying web server.

To mitigate this risk, follow the steps given

1. Delete the examples application already deployed on the tomcat server.

E.g. webapps/examples