



Informatica® Cloud Application Integration
September 2024

Loan Processing with Azure OpenAI

© Copyright Informatica LLC 2024

This software and documentation contain proprietary information of Informatica LLC and are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright law. Reverse engineering of the software is prohibited. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC. This Software may be protected by U.S. and/or international Patents and other Patents Pending.

Use, duplication, or disclosure of the Software by the U.S. Government is subject to the restrictions set forth in the applicable software license agreement and as provided in DFARS 227.7202-1(a) and 227.7702-3(a) (1995), DFARS 252.227-7013(1)(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable.

The information in this product or documentation is subject to change without notice. If you find any problems in this product or documentation, please report them to us in writing.

Informatica, Informatica Platform, Informatica Data Services, PowerCenter, PowerCenterRT, PowerCenter Connect, PowerCenter Data Analyzer, PowerExchange, PowerMart, Metadata Manager, Informatica Data Quality, Informatica Data Explorer, Informatica B2B Data Transformation, Informatica B2B Data Exchange Informatica On Demand, Informatica Identity Resolution, Informatica Application Information Lifecycle Management, Informatica Complex Event Processing, Ultra Messaging, Informatica Master Data Management, and Live Data Map are trademarks or registered trademarks of Informatica LLC in the United States and in jurisdictions throughout the world. All other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties, including without limitation: Copyright DataDirect Technologies. All rights reserved. Copyright © Sun Microsystems. All rights reserved. Copyright © RSA Security Inc. All Rights Reserved. Copyright © Ordinal Technology Corp. All rights reserved. Copyright © Aandacht c.v. All rights reserved. Copyright Genivia, Inc. All rights reserved. Copyright Isomorphic Software. All rights reserved. Copyright © Meta Integration Technology, Inc. All rights reserved. Copyright © Intalio. All rights reserved. Copyright © Oracle. All rights reserved. Copyright © Adobe Systems Incorporated. All rights reserved. Copyright © DataArt, Inc. All rights reserved. Copyright © ComponentSource. All rights reserved. Copyright © Microsoft Corporation. All rights reserved. Copyright © Rogue Wave Software, Inc. All rights reserved. Copyright © Teradata Corporation. All rights reserved. Copyright © Yahoo! Inc. All rights reserved. Copyright © Glyph & Cog, LLC. All rights reserved. Copyright © Thinkmap, Inc. All rights reserved. Copyright © Clearpace Software Limited. All rights reserved. Copyright © Information Builders, Inc. All rights reserved. Copyright © OSS Nokalva, Inc. All rights reserved. Copyright Edifecs, Inc. All rights reserved. Copyright Cleo Communications, Inc. All rights reserved. Copyright © International Organization for Standardization 1986. All rights reserved. Copyright © ej-technologies GmbH. All rights reserved. Copyright © Jaspersoft Corporation. All rights reserved. Copyright © International Business Machines Corporation. All rights reserved. Copyright © yWorks GmbH. All rights reserved. Copyright © Lucent Technologies. All rights reserved. Copyright © University of Toronto. All rights reserved. Copyright © Daniel Veillard. All rights reserved. Copyright © Unicode, Inc. Copyright IBM Corp. All rights reserved. Copyright © MicroQuill Software Publishing, Inc. All rights reserved. Copyright © PassMark Software Pty Ltd. All rights reserved. Copyright © LogiXML, Inc. All rights reserved. Copyright © 2003-2010 Lorenzi Davide, All rights reserved. Copyright © Red Hat, Inc. All rights reserved. Copyright © The Board of Trustees of the Leland Stanford Junior University. All rights reserved. Copyright © EMC Corporation. All rights reserved. Copyright © Flexera Software. All rights reserved. Copyright © Jinfonet Software. All rights reserved. Copyright © Apple Inc. All rights reserved. Copyright © Teleric Inc. All rights reserved. Copyright © BEA Systems. All rights reserved. Copyright © PDFlib GmbH. All rights reserved. Copyright © Orientation in Objects GmbH. All rights reserved. Copyright © Tanuki Software, Ltd. All rights reserved. Copyright © Ricebridge. All rights reserved. Copyright © Sencha, Inc. All rights reserved. Copyright © Scalable Systems, Inc. All rights reserved. Copyright © jQWidgets. All rights reserved. Copyright © Tableau Software, Inc. All rights reserved. Copyright © MaxMind, Inc. All Rights Reserved. Copyright © TMate Software s.r.o. All rights reserved. Copyright © MapR Technologies Inc. All rights reserved. Copyright © Amazon Corporate LLC. All rights reserved. Copyright © Highsoft. All rights reserved. Copyright © Python Software Foundation. All rights reserved. Copyright © BeOpen.com. All rights reserved. Copyright © CNRI. All rights reserved.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>), and/or other software which is licensed under various versions of the Apache License (the "License"). You may obtain a copy of these Licenses at <http://www.apache.org/licenses/>. Unless required by applicable law or agreed to in writing, software distributed under these Licenses is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the Licenses for the specific language governing permissions and limitations under the Licenses.

This product includes software which was developed by Mozilla (<http://www.mozilla.org/>), software copyright The JBoss Group, LLC, all rights reserved; software copyright © 1999-2006 by Bruno Lowagie and Paulo Soares and other software which is licensed under various versions of the GNU Lesser General Public License Agreement, which may be found at <http://www.gnu.org/licenses/lgpl.html>. The materials are provided free of charge by Informatica, "as-is", without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

The product includes ACE(TM) and TAO(TM) software copyrighted by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University, Copyright (©) 1993-2006, all rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (copyright The OpenSSL Project. All Rights Reserved) and redistribution of this software is subject to terms available at <http://www.openssl.org> and <http://www.openssl.org/source/license.html>.

This product includes Curl software which is Copyright 1996-2013, Daniel Stenberg, <daniel@haxx.se>. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://curl.haxx.se/docs/copyright.html>. Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

The product includes software copyright 2001-2005 (©) MetaStuff, Ltd. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://www.dom4j.org/license.html>.

The product includes software copyright © 2004-2007, The Dojo Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://dojotoolkit.org/license>.

This product includes ICU software which is copyright International Business Machines Corporation and others. All rights reserved. Permissions and limitations regarding this software are subject to terms available at <http://source.icu-project.org/repos/icu/icu/trunk/license.html>.

This product includes software copyright © 1996-2006 Per Bothner. All rights reserved. Your right to use such materials is set forth in the license which may be found at <http://www.gnu.org/software/kawa/Software-License.html>.

This product includes OSSP UUID software which is Copyright © 2002 Ralf S. Engelschall, Copyright © 2002 The OSSP Project Copyright © 2002 Cable & Wireless Deutschland. Permissions and limitations regarding this software are subject to terms available at <http://www.opensource.org/licenses/mit-license.php>.

This product includes software developed by Boost (<http://www.boost.org/>) or under the Boost software license. Permissions and limitations regarding this software are subject to terms available at http://www.boost.org/LICENSE_1_0.txt.

This product includes software copyright © 1997-2007 University of Cambridge. Permissions and limitations regarding this software are subject to terms available at <http://www.pcre.org/license.txt>.

This product includes software copyright © 2007 The Eclipse Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://www.eclipse.org/org/documents/epl-v10.php> and at <http://www.eclipse.org/org/documents/edl-v10.php>.

This product includes software licensed under the terms at <http://www.tcl.tk/software/tcltk/license.html>, <http://www.bosrup.com/web/overlib/?License>, <http://www.stlport.org/doc/license.html>, <http://asm.ow2.org/license.html>, <http://www.cryptix.org/LICENSE.TXT>, <http://hsqldb.org/web/hsqldbLicense.html>, <http://httpunit.sourceforge.net/doc/license.html>, <http://jung.sourceforge.net/license.txt>, http://www.gzip.org/zlib/zlib_license.html, <http://www.openldap.org/software/release/license.html>, <http://www.libssh2.org>, <http://slf4j.org/license.html>, <http://www.sente.ch/software/OpenSourceLicense.html>, <http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3-license-agreement>; <http://antlr.org/license.html>; <http://aopalliance.sourceforge.net/>; <http://www.bouncycastle.org/licence.html>; <http://www.jgraph.com/jgraphdownload.html>; <http://www.jcraft.com/jsch/LICENSE.txt>; http://jotm.objectweb.org/bsd_license.html; <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>; <http://www.slf4j.org/license.html>; <http://nanoxml.sourceforge.net/orig/copyright.html>; <http://www.json.org/license.html>; <http://forge.ow2.org/projects/javaservice/>; <http://www.postgresql.org/about/license.html>, <http://www.sqlite.org/copyright.html>, <http://www.tcl.tk/software/tcltk/license.html>, <http://www.jaxen.org/faq.html>, <http://www.jdom.org/docs/faq.html>, <http://www.slf4j.org/license.html>; <http://www.iodbc.org/dataspace/iodbc/wiki/IODBC/License>; <http://www.keplerproject.org/md5/license.html>; <http://www.toedter.com/en/jcalendar/license.html>; <http://www.edankert.com/bounce/index.html>; <http://www.net-snmp.org/about/license.html>; <http://www.openmdx.org/#FAQ>; http://www.php.net/license/3_01.txt; <http://srp.stanford.edu/license.txt>; <http://www.schneier.com/blowfish.html>; <http://www.jmock.org/license.html>; <http://xsom.java.net>; <http://benalman.com/about/license/>; <https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js>; <http://www.h2database.com/html/license.html#summary>; <http://jsoncpp.sourceforge.net/LICENSE>; <http://jdbc.postgresql.org/license.html>; <http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto>; <https://github.com/rantav/hector/blob/master/LICENSE>; <http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html>; <http://jibx.sourceforge.net/jibx-license.html>; <https://github.com/lyokato/libgeohash/blob/master/LICENSE>; <https://github.com/hjiang/jsonxx/blob/master/LICENSE>; <https://code.google.com/p/lz4/>; <https://github.com/jedisct1/libsodium/blob/master/LICENSE>; <http://one-jar.sourceforge.net/index.php?page=documents&file=license>; <https://github.com/EsotericSoftware/kryo/blob/master/license.txt>; <http://www.scala-lang.org/license.html>; <https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt>; <http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html>; <https://aws.amazon.com/asl/>; <https://github.com/twbs/bootstrap/blob/master/LICENSE>; <https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt>; <https://github.com/documentcloud/underscore-contrib/blob/master/LICENSE>, and <https://github.com/apache/hbase/blob/master/LICENSE.txt>.

This product includes software licensed under the Academic Free License (<http://www.opensource.org/licenses/afl-3.0.php>), the Common Development and Distribution License (<http://www.opensource.org/licenses/cddl1.php>), the Common Public License (<http://www.opensource.org/licenses/cpl1.0.php>), the Sun Binary Code License Agreement Supplemental License Terms, the BSD License (<http://www.opensource.org/licenses/bsd-license.php>), the new BSD License (<http://opensource.org/licenses/BSD-3-Clause>), the MIT License (<http://www.opensource.org/licenses/mit-license.php>), the Artistic License (<http://www.opensource.org/licenses/artistic-license-1.0>) and the Initial Developer's Public License Version 1.0 (<http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/>).

This product includes software copyright © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. All rights reserved. Permissions and limitations regarding this software are subject to terms available at <http://xstream.codehaus.org/license.html>. This product includes software developed by the Indiana University Extreme! Lab. For further information please visit <http://www.extreme.indiana.edu/>.

This product includes software Copyright (c) 2013 Frank Balluffi and Markus Moeller. All rights reserved. Permissions and limitations regarding this software are subject to terms of the MIT license.

See patents at <https://www.informatica.com/legal/patents.html>.

DISCLAIMER: Informatica LLC provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.
2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

Publication Date: 2024-09-30

Table of Contents

- Preface 5**

- Chapter 1: Introduction to Loan Processing with Azure OpenAI recipe..... 6**
 - Loan Processing with Azure OpenAI recipe contents. 6
 - Loan Processing with Azure OpenAI recipe assets. 7

- Chapter 2: Using the Loan Processing with Azure OpenAI recipe..... 9**
 - Step 1: Copy and access the recipe. 9
 - Step 2: Publish the DocuSignConnector service connector 10
 - Step 3: Configure and publish the Email connection. 10
 - Step 4: Configure and publish the DocuSign connection. 12
 - Step 5: Configure and publish the Azure OpenAI Connection. 13
 - Step 6: Publish the processes 13
 - Step 7: Invoke the process 14
 - Publishing and running the Guide. 14
 - REST or SOAP API endpoints. 14
 - Run Using. 16

Preface

Use *Loan Processing with Azure OpenAI* to learn how to evaluate a loan request and approve or reject it based on the applicant's credit score. This guide assumes that you have an understanding of the DocuSign Connector, Email Connector, and Azure OpenAI Connector concepts.

CHAPTER 1

Introduction to Loan Processing with Azure OpenAI recipe

The Loan Processing with Azure OpenAI recipe is based on REST and SOAP APIs. The process is called by an HTTP request with basic information about the loan request as an incoming parameter.

The process performs initial verification by validating the loan information. After successful validation, the process sends an email for application submission, generates a loan ID, and verifies employment details and income.

The process then performs a credit check to assess any risk by calling the LLM and summarizing the application. If the credit score of the applicant is more than 600, the loan gets automatically approved and an approval email is sent to the applicant with the document for an e-signature. Otherwise, an email is sent to the applicant stating that the loan application is under review.

Simultaneously, an email is sent to the reviewer to review the loan request. Based on the reviewer's decision, if the loan is approved, an approval email is sent to the applicant with the document for an e-signature. Otherwise, a loan rejection email is sent.

Loan Processing with Azure OpenAI recipe contents

The Loan Processing with Azure OpenAI recipe contains process objects, app connections, processes, a human task, a service connector, and a guide.

The following image shows the assets that the Loan Processing with Azure OpenAI recipe package contains:

| Name | Type | Updated On | Description | Tags | Status | Published |
|--|----------------|---------------|---|------|--------|-----------|
| Address | Process Object | Sep 26, 20... | Process object that provides address details for loan processing | | Valid | |
| LoanRequest | Process Object | Sep 26, 20... | Process object that provides the required fields for loan processing | | Valid | |
| EmploymentDetails | Process Object | Sep 26, 20... | Process object that provides employment details needed for loan processing | | Valid | |
| SettingsLLM | Process Object | Sep 30, 20... | Process object that creates the LLM settings for Azure OpenAI | | Valid | |
| PersonalDetails | Process Object | Sep 26, 20... | Process object that provides personal details required for loan processing | | Valid | |
| Risk Assessment - LLM using Azure OpenAI | Process | Sep 27, 20... | Subprocess that calls an ML model to determine the associated risk. | | Valid | Published |
| Send Document for eSignature | Process | Sep 26, 20... | Subprocess to upload documents to DocuSign for signatures and wait for completion | | Valid | Published |
| Credit Check | Process | Sep 30, 20... | Subprocess to run a credit check | | Valid | Published |
| Process Loan Request LLM | Process | Sep 30, 20... | The process is called by an HTTP request with basic information about the loan request as an incomin... | | Valid | Published |
| Validate Loan Information | Process | Sep 26, 20... | Subprocess to validate loan information requests | | Valid | Published |
| Validate Employment | Process | Sep 26, 20... | Subprocess to validate employment information for an applicant | | Valid | Published |
| ApproveLoan | Human Task | Sep 26, 20... | Human Task that reviews the loan request for risk | | Valid | |
| Loan Processor | Guide | Sep 30, 20... | Guide that demonstrates loan processing with Azure OpenAI | | Valid | Published |
| DocuSignConnector | Service Con... | Sep 26, 20... | Connector that exposes various actions of DocuSign | | Valid | Published |
| EmailConnection | App Connect... | Sep 26, 20... | Email connection to send an email | | Valid | Published |
| DocuSignConnection | App Connect... | Sep 26, 20... | Connection that calls the DocuSign REST API | | Valid | Published |
| AzureOpenAIConnection | App Connect... | Sep 26, 20... | Azure OpenAI connection that connects to Azure OpenAI and performs actions, such as generating c... | | Valid | Published |

Loan Processing with Azure OpenAI recipe assets

The following table lists the assets that the Loan Processing with Azure OpenAI recipe package contains:

| Asset Name | Asset Type | Description |
|-----------------------|-------------------|--|
| Address | Process object | Provides address details for loan processing. |
| LoanRequest | Process object | Provides the required fields for loan processing. |
| SettingsLLM | Process object | Creates the LLM settings for Azure OpenAI. |
| PersonalDetails | Process object | Provides personal details for loan processing. |
| EmploymentDetails | Process object | Provides employment details for loan processing. |
| DocuSignConnector | Service connector | Provides various actions to perform on DocuSign. |
| EmailConnection | App connection | Email connection that is used to send emails. |
| DocuSignConnection | App connection | Calls the DocuSign REST API. |
| AzureOpenAIConnection | App connection | Connects to AzureOpenAI and performs actions, such as generating content, counting tokens, and listing models. |

| Asset Name | Asset Type | Description |
|--|------------|---|
| ApproveLoan | Human task | Reviews the loan request for risk. To work with this step, you must have the human task feature enabled for your organization. For more information, see <i>Design</i> . |
| Validate Employment | Process | Subprocess to validate employment information for an applicant. |
| Validate Loan Information | Process | Subprocess to validate loan information requests. |
| Risk Assessment - LLM using Azure OpenAI | Process | Subprocess that calls an ML model to determine the associated risk. |
| Send Document for eSignature | Process | Subprocess to upload documents to DocuSign for signatures and wait for completion. |
| Credit Check | Process | <p>Subprocess to run a credit check. The credit score is set to 600, by default. If the applicant's credit score is more than 600, the loan is approved automatically. Otherwise, the application is sent to the reviewer for approval.</p> <p>You can update the logic to get the credit history and credit score in real time by using credit unions such as Equifax and TransUnion as needed and modify the rules on when the loan must be approved or needs review.</p> |
| Process Loan Request LLM | Process | <p>The process is called by an HTTP request with basic information about the loan request as an incoming parameter. The process performs initial verification by validating the loan information. After successful validation, the process sends an email for application submission, generates a loan ID, and verifies employment details and income.</p> <p>The process then performs a credit check to assess any risk by calling the LLM and summarizing the application. If the credit score of the applicant is more than 600, the loan gets automatically approved and an approval email is sent to the applicant with the document for an e-signature. Otherwise, an email is sent to the applicant stating that the loan application is under review.</p> <p>Simultaneously, an email is sent to the reviewer to review the loan request. Based on the reviewer's decision, if the loan is approved, an approval email is sent to the applicant with the document for an e-signature. Otherwise, a loan rejection email is sent.</p> |
| Loan Processor | Guide | Demonstrates loan processing with AzureOpenAI. |

CHAPTER 2

Using the Loan Processing with Azure OpenAI recipe

To use the Loan Processing with Azure OpenAI recipe, you must perform the following steps manually:

Step 1: Copy and access the recipe

Step 2: Publish the DocuSignConnector service connector

Step 3: Configure and publish the Email connection

Step 4: Configure and publish the DocuSign connection

Step 5: Configure and publish the Azure OpenAI connection

Step 6: Publish the processes

Step 7: Invoke the process

Step 1: Copy and access the recipe

Copy the pre-configured assets in the recipe to a separate project or folder.

1. Open the **Loan Processing with Azure OpenAI** recipe and click **Use**.
2. Select the location where you want to copy the recipe, and then click **Continue**.
3. In the **Copying the recipe** dialog box, click **OK**.

It might take some time for the recipe to get copied. You will receive a notification when the recipe is ready for use.

4. After the recipe is copied, click **Explore** to access the recipe content.

- Navigate to the project or folder where you copied the recipe or enter the recipe name in the **Find** box. All the assets in the recipe are displayed as shown in the following image:

| Name | Type | Updated On | Description | Tags | Status | Published |
|--|----------------|---------------|---|------|--------|-----------|
| Address | Process Object | Sep 26, 20... | Process object that provides address details for loan processing | | Valid | |
| LoanRequest | Process Object | Sep 26, 20... | Process object that provides the required fields for loan processing | | Valid | |
| EmploymentDetails | Process Object | Sep 26, 20... | Process object that provides employment details needed for loan processing | | Valid | |
| SettingsLLM | Process Object | Sep 30, 20... | Process object that creates the LLM settings for Azure OpenAI | | Valid | |
| PersonalDetails | Process Object | Sep 26, 20... | Process object that provides personal details required for loan processing | | Valid | |
| Risk Assessment - LLM using Azure OpenAI | Process | Sep 27, 20... | Subprocess that calls an ML model to determine the associated risk. | | Valid | Published |
| Send Document for eSignature | Process | Sep 26, 20... | Subprocess to upload documents to DocuSign for signatures and wait for completion | | Valid | Published |
| Credit Check | Process | Sep 30, 20... | Subprocess to run a credit check | | Valid | Published |
| Process Loan Request LLM | Process | Sep 30, 20... | The process is called by an HTTP request with basic information about the loan request as an incomin... | | Valid | Published |
| Validate Loan Information | Process | Sep 26, 20... | Subprocess to validate loan information requests | | Valid | Published |
| Validate Employment | Process | Sep 26, 20... | Subprocess to validate employment information for an applicant | | Valid | Published |
| ApproveLoan | Human Task | Sep 26, 20... | Human Task that reviews the loan request for risk | | Valid | |
| Loan Processor | Guide | Sep 30, 20... | Guide that demonstrates loan processing with Azure OpenAI | | Valid | Published |
| DocuSignConnector | Service Con... | Sep 26, 20... | Connector that exposes various actions of DocuSign | | Valid | Published |
| EmailConnection | App Connect... | Sep 26, 20... | Email connection to send an email | | Valid | Published |
| DocuSignConnection | App Connect... | Sep 26, 20... | Connection that calls the DocuSign REST API | | Valid | Published |
| AzureOpenAIConnection | App Connect... | Sep 26, 20... | Azure OpenAI connection that connects to Azure OpenAI and performs actions, such as generating c... | | Valid | Published |

Step 2: Publish the DocuSignConnector service connector

To publish the **DocuSignConnector** service connector, open the **DocuSignConnector** service connector and click **Publish**.

Step 3: Configure and publish the Email connection

Configure the authentication details in the EmailConnection connection, and then publish the connection.

- Open the **EmailConnection** connection.
- From the **Type** list, select **IICS Cloud Application Integration Email Service (Licensed for use)**.
- From the **Run On** list, select **Cloud Server or any Secure Agent**.

4. From the **Authentication Type** list, select **Password** or **OAuth** as needed. Based on the authentication type selected, perform one of the following steps:

- For **Password** authentication, enter values for the following properties in the **Connection Properties** section:

| Property | Description |
|----------------|---|
| Authentication | Select Enable . Email Connector authenticates the user name and password that you enter in the email connection properties. |
| User Name | User name to log in to the email server. The user name is either the account name or the email address that is used to send the email with the synchronization results. For example: <code>notifyme@mydomain.com</code> |
| Password | Password for the email address. Set an API key for your email account. For information about creating an API key, see Create API credentials . |
| Security | Select SSL for the Email connection to use the SSL protocol. |

Configure the following common properties on the connection creation page:

| Property | Description |
|----------|--|
| Host | Email server's DNS name, such as <code>mail.mydomain.com</code> , or an IP address, such as <code>192.168.1.1</code> . |
| Port | Port for communication between the Process Server and the email server. Default is 25 . |

The following image shows the **EmailConnection** connection detail page with the authentication type set to **Password**:

- For **OAuth** authentication, enter values for the following properties in the **Connection Properties** section:

| Property | Description |
|-------------------|--|
| Authorization URL | Enter the OAuth authorization URL for the email service that is used to authorize the user request. For example: <code>https://login.microsoftonline.com/xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxx/oauth2/v2.0/authorize</code> |
| Token Request URL | Enter the OAuth token request URL that handles token requests. For example: <code>https://login.microsoftonline.com/xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxx/oauth2/v2.0/token</code> The refresh token expires in 90 days. The user must authenticate again and publish the connection before the token expires. |
| Client ID | Specify the identifier value from the OAuth provider. |

| Property | Description |
|---------------|---|
| Client Secret | Enter the client secret to connect to the email application. |
| Scope | Specify the scope. The scope in OAuth authentication limits an application's access to a user's account. You can select multiple scopes for a single client. To enter multiple scopes, separate each value with a space. For a Microsoft Outlook email account, enter the following scope: <code>https://outlook.office.com/SMTP.Send offline_access</code> |

Configure the following common properties on the connection creation page:

| Property | Description |
|----------|--|
| Host | Email server's DNS name, such as <code>mail.mydomain.com</code> , or an IP address, such as <code>192.168.1.1</code> . |
| Port | Port for communication between the Process Server and the email server. Default is 25 . |

5. Save and publish the connection.

Step 4: Configure and publish the DocuSign connection

Configure the authentication details to connect to DocuSign in the DocuSignConnection connection, and then publish the connection.

1. Open the **DocuSignConnection** connection.
2. From the **Type** list, select **DocuSignConnector** from the recipe assets folder.
3. From the **Run On** list, select **Cloud Server or any Secure Agent**.
4. In the **Connection Properties** section, enter values for the following properties:

| Property | Description |
|-----------|--|
| AccountID | DocuSign account ID. |
| Username | User name to log in to the DocuSign account. |
| Password | Password to log in to the DocuSign account. |
| APIkey | Authenticates DocuSign connection requests. |
| BaseURL | URL to access the APIs in DocuSign. |

5. Save and publish the connection.

Step 5: Configure and publish the Azure OpenAI Connection

Configure the endpoint URL and API key in the AzureOpenAIConnection connection, and then publish the connection.

1. Open the **AzureOpenAIConnection** connection.
2. In the **Connection Properties** section, enter values for the following properties:

| Property | Description |
|--------------|--|
| Endpoint_URL | The REST API endpoint for Azure OpenAI. You can find this value in the Keys & Endpoint section when examining your resource from the Azure portal. Alternatively, you can find the value in Azure OpenAI Studio > Playground > Code View . |
| API_Key | The API key to authenticate Azure OpenAI connection requests. You can find this value in the Keys & Endpoint section when examining your resource from the Azure portal. Alternatively, you can find the value in Azure OpenAI Studio > Playground > Code View . |

3. Save and publish the connection.

Step 6: Publish the processes

1. Open the following processes in the order specified below:
 - a. Validate Employment
 - b. Validate Loan Information
 - c. Risk Assessment - LLM using Azure OpenAI
 - d. Send Document for eSignature
 - e. Credit Check
2. Save and publish all the processes.
3. Open the **Process Loan Request LLM** process.
4. In the **Set LLM properties** step, enter the instructions in the **Assignments** field by updating the **LLM_Settings** field using the Expression Editor, as shown in the following sample code:

```
if (exists($input.LLM_Settings))
then $input.LLM_Settings
else <SettingsLLM>
  <Top_P>1</Top_P>
  <Temperature>0.5</Temperature>
  <Max_Tokens>500</Max_Tokens>
  <Deployment_ID><Deployment ID of the LLM model></Deployment_ID>
  <API_Version><API version of the deployed LLM model></API_Version>
</SettingsLLM>
```

To invoke the process using the guide or the run using option, you must configure these properties in the Set LLM properties step. To invoke the process using the REST or SOAP API endpoints, you can optionally provide these properties as input into the request payload.

5. Save and publish the process.

Step 7: Invoke the process

You can invoke the Process Loan Request LLM process to evaluate a loan request and approve or reject it based on the applicant's credit score by using one of the following options:

- [“Publishing and running the Guide” on page 14](#)
- [“REST or SOAP API endpoints” on page 14](#)
- [“Run Using” on page 16](#)

Publishing and running the Guide

Publish and run the guide and enter the required details on the screens.

1. Open the **Loan Processor** guide.
2. On the **Start** tab of the Start step, ensure that the **Run As** field is set to **Current User**.
3. Save and publish the guide.
4. On the **Actions** menu, click **Run**. Alternatively, you can copy the execution URL from the **Properties Details** dialog box to run the guide.
5. On the **Instructions** page, enter information related to personal details, employment details, and loan details, and attach an identity proof. The LLM uses this information to perform initial verification.
6. Click **Submit Loan Request**.
If the credit score of the applicant is more than 600, the loan gets automatically approved and an approval email is sent to the applicant with the document for an e-signature. Otherwise, an email is sent to the applicant stating that the loan application is under review. Simultaneously, an email is sent to the reviewer to review the loan request. Based on the reviewer's decision, if the loan is approved, an approval email is sent to the applicant with the document for an e-signature. Otherwise, a loan rejection email is sent.
7. On the next screen, the load ID and loan status appear.
8. Click **Continue**.
9. Click **Done**.

You can also use the embed code to embed the guide into an HTML document of a third-party application.

Note: If you encounter any issue with loading the guide, check whether you are using the default guide theme.

REST or SOAP API endpoints

Pass input through REST or SOAP API endpoints in any API client such as cURL, Postman, SOAP UI, or through any programming language.

For example:

1. Open Postman.
2. Select the HTTP verb as POST and specify the generated REST service URL and payload as shown in the following image:

https://pod1-cai.mrel.infaqa.com/active-bpel/public/rt/4NpEu500PycjzMHq05p8G7/Process_Loan_Request_LLM

POST https://pod1-cai.mrel.infaqa.com/active-bpel/public/rt/4NpEu500PycjzMHq05p8G7/Process_Loan_Request_LLM

Params Authorization Headers (9) **Body** Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL **JSON** Beautify

```

1 {
2   "LoanRequest": {
3     "employmentDetails": {
4       "jobTitle": "Developer",
5       "employerName": "PM",
6       "monthlyIncome": "10000"
7     },
8     "customerID": "12354",
9     "loanPurpose": "home improvement",
10    "personalDetails": {
11      "lastName": "Pmd",
12      "firstName": "FirstName",
13      "address": {
14        "city": "Texas",
15        "street": "Street",
16        "postalcode": "563210",
17        "state": "State"
18      },
19      "phone": "9999999999",
20      "email": "pram@gmail.com"
21    },
22    "loanAmount": "2000",
23    "ssn": "12323274387823"
24  },
25  "LLM_Settings":
26  {
27    "Top_P": "1",
28    "Temperature": "0.9",
29    "Max_Tokens": "300",
30    "Deployment_ID": "cai-qa-gpt4turbo-sweden",
31    "API_Version": "2024-06-01"
32  }
33 }
34

```

Body Cookies Headers (18) Test Results 200 OK • 15.19 s • 693 B Save Response

Here is the sample payload:

```

{
  "LoanRequest": {
    "employmentDetails": {
      "jobTitle": "Developer",
      "employerName": "PM",
      "monthlyIncome": "10000"
    },
    "customerID": "12354",
    "loanPurpose": "home improvement",
    "personalDetails": {
      "lastName": "Pmd",
      "firstName": "FirstName",
      "address": {
        "city": "Texas",
        "street": "Street",
        "postalcode": "563210",
        "state": "State"
      },
      "phone": "9999999999",
      "email": "pram@gmail.com"
    },
    "loanAmount": "2000",
    "ssn": "12323274387823"
  },
  "LLM_Settings":
  {
    "Top_P": "1",
    "Temperature": "0.9",
    "Max_Tokens": "300",
    "Deployment_ID": "cai-qa-gpt4turbo-sweden",
    "API_Version": "2024-06-01"
  }
}

```

```
}  
}
```

Optionally, you can configure the LLM settings properties in the **Set LLM properties** step in the Process Loan Request LLM process. For more information, see [“Step 6: Publish the processes ” on page 13](#). If the LLM settings properties are preconfigured in the Process Loan Request LLM process and you also enter them as input in the request payload, the request payload values take precedence.

3. Enter the user account details on the **Authorization** tab.
4. Click **Send**.

Run Using

Pass input in the JSON or XML format through the **Run Using** option of the process.

1. Open the **Process Loan Request LLM** process and click **Actions > Run Using**.
2. Click **New Input**, enter a name for the process input, and click **Save**.
The payload is populated in the **Process Input** section.
3. From the **Encoding** list, select **JSON** or **XML** based on the format that you want to work with.
4. Enter the values for the process object fields in the payload.
5. Validate, save, and run the process.