MyPureCloud / **purecloud-for-salesforce-einstein-example**

This project demonstrates how to use an endpoint exposed in PureCloud for Salesforce to receive incoming chat messages, search Salesforce Einstein for the intent, and use that intent to search and display relevant Salesforce Knowledge articles in a Salesforce Lightning component. The project includes a sample dataset to train Salesforce Einstein...

MIT License

2 stars 2 forks

Star Watch

Code Issues Pull requests Actions Projects Wiki Security

master

ININDevEvangelists Updating to standard MIT license ... on Feb 8 20

View code

README.md

PureCloud for Salesforce Einstein Example

This repository contains a Salesforce Lightning component, Apex classes, and supporting files for use with PureCloud for Salesforce. These items show how Salesforce Einstein can display Salesforce Knowledge articles based on ACD chat messages in PureCloud for Salesforce.

⚠️ **Warning**: The Lightning component code contained in this repository depends on Lightning Message Service (LMS), which is currently a beta feature in Salesforce. The Lightning component code will not work in the Lightning Experience example until the following occurs:

- Salesforce announces the general availability of LMS.
- Genesys updates and releases the PureCloud for Salesforce managed package with LMS support.

Until then, use the post message example branch for a working example.

Table of Contents

Getting Started

TL;DR

1. Set up Salesforce Knowledge and train your data for Salesforce Einstein.
2. Configure the PureCloud for Salesforce managed package to enable client events with an event type of Notification and expand chat notifications.
3. Install the unmanaged package, add a Lightning component to an app, and add values to the PureCloudKnowledgeConstants Apex class.
4. Send an ACD chat message to an active user in PureCloud for Salesforce and confirm that the Lightning component updates with articles.

Prerequisites

- PureCloud for Salesforce installed in your Salesforce organization. For more information, see Set up the PureCloud for Salesforce integration.
  - Version of the PureCloud for Salesforce managed package that supports Lightning Message Service.
- ACD chat working in your PureCloud organization.

Installation

Install the unmanaged package.

Configuration

Set Up Salesforce Knowledge

Important: If the name of your knowledge article object is different from the default name (Knowledge__kav), you will not see any search results. To fix the problem, replace every instance of "Knowledge__kav" in PureCloudKnowledgeUtilityController.apxc with your object name.

1. Enable Salesforce Knowledge.
2. Import Salesforce Knowledge articles from the simple-sample-knowledge-articles.csv file.
   You can import articles one of two ways:
   - Use Import Articles in Salesforce.
   - Download Salesforce's Data Loader application.
3. Publish the articles.

Sign Up for Salesforce Einstein

1. Go to Einstein Platform Services.
2. Click **Sign Up Using Salesforce**.
3. On the activation page, download the **einstein_platform.pem** file. The file contains your key.
4. Check your email to verify your new account.

Train Salesforce Einstein

These steps use the example dataset in the einstein-example-dataset copy.csv file. The following API calls and information come from the Salesforce Einstein documentation.

1. Generate an OAuth token to use in the API calls. a. Go to Generate an OAuth token. b. Enter your credentials for Einstein and the key from the **einstein_platform.pem** file.
2. Create the dataset using the **einstein-example-dataset copy.csv** file.

```
  curl -X POST -H "Authorization: Bearer  <TOKEN>" -H "Cache-Control: no-
cache" -H "Content-Type: multipart/form-data" -F "type=text-intent" -F "name=
<DATASET_NAME>" -F "data=@<FILE_LOCATION>"
https://api.einstein.ai/v2/language/datasets/upload
```

This call is asynchronous. You receive a dataset ID that you can use to check the availability of the dataset.

```
curl -X GET -H "Authorization: Bearer  <TOKEN>" -H "Cache-Control: no-cache"
https://api.einstein.ai/v2/language/datasets/<DATASET_ID>
```

3. Train the dataset and save the model ID that is returned.

```
curl -X POST -H "Authorization: Bearer <TOKEN>" -H "Cache-Control: no-cache"
-H "Content-Type: multipart/form-data" -F "name=<DATASET_NAME>" -F
"datasetId=<DATASET_ID>" https://api.einstein.ai/v2/language/train
```

This call is asynchronous. You receive a model ID that you can use to check the status of the training.

```
curl -X GET -H "Authorization: Bearer  <TOKEN>" -H "Cache-Control: no-cache"
https://api.einstein.ai/v2/language/train/<MODEL_ID>
```

The training is complete when status is SUCCEEDED and progress is 1.

Configure the PureCloud for Salesforce Managed Package

1. In your Salesforce organization, click **Configure** next to the PureCloud for Salesforce managed package.
2. Under **Choose a Call Center**, select **PureCloud for Salesforce Lightning**.
3. Select **Enable Client Events**.
4. Under **Client Event Types**, select **Notification**. Click the right arrow to add it under **Chosen**.
5. Select **Expand Chat Notifications**.
6. Click **Save.**

Install the Unmanaged Package

1. In your Salesforce organization, install the unmanaged package.
2. Add a **Remote Site Setting**. Set **Remote Site URL** to `https://api.einstein.ai` .
3. (Optional) To dynamically create access tokens, use Salesforce Files to upload the **einstein_platform.pem** file. You downloaded this file when you signed up for Salesforce Einstein.
   a. Add the **Files** object as a navigation item to a Lighting app.
   b. Open a Lightning app and click **Files** in the top toolbar. Then click **Upload Files**.

4. Add values to the **PureCloudKnowledgeConstants** Apex class. At a minimum, add **USER_EMAIL** and **MODEL_ID**.
   If you did not upload the **einstein_platform.pem** file in step 3, then also add **ACCESS_TOKEN**. The unmanaged package uses this access token.
5. Add the **PureCloudKnowledgeUtility** Lightning component as a utility item to a Lightning app. This Lightning app must have an Open CTI Softphone.
   Note: Be sure to select **Start automatically**.

Usage

1. Send an ACD chat message to an active user in PureCloud for Salesforce.
2. From the end user's chat window, type a message that corresponds with the dataset that was used to train Einstein.
3. Open the Lightning component. The Lightning component updates with articles related to the chat message.
4. Click an article. Confirm that the article opens inside Salesforce.

Additional Information

Releases

No releases published

Packages

No packages published

Contributors 5

Languages

JavaScript 87.2% CSS 12.8%