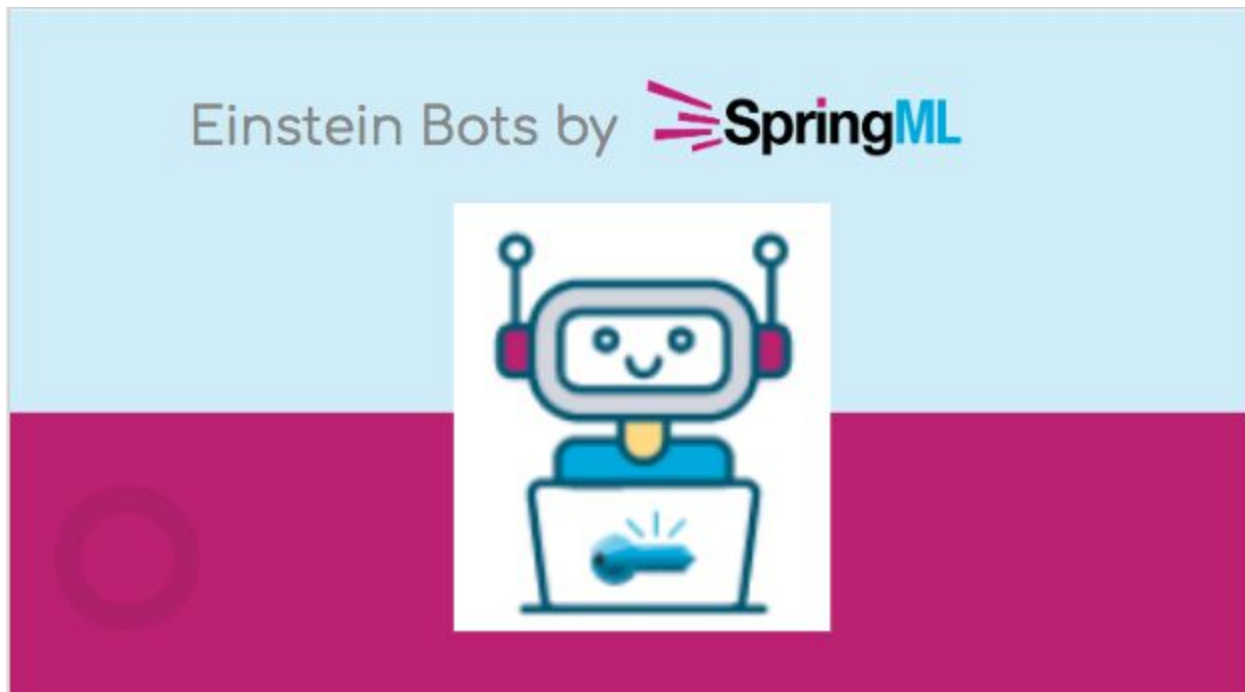


HLS Payers Einstein Bot - Installation

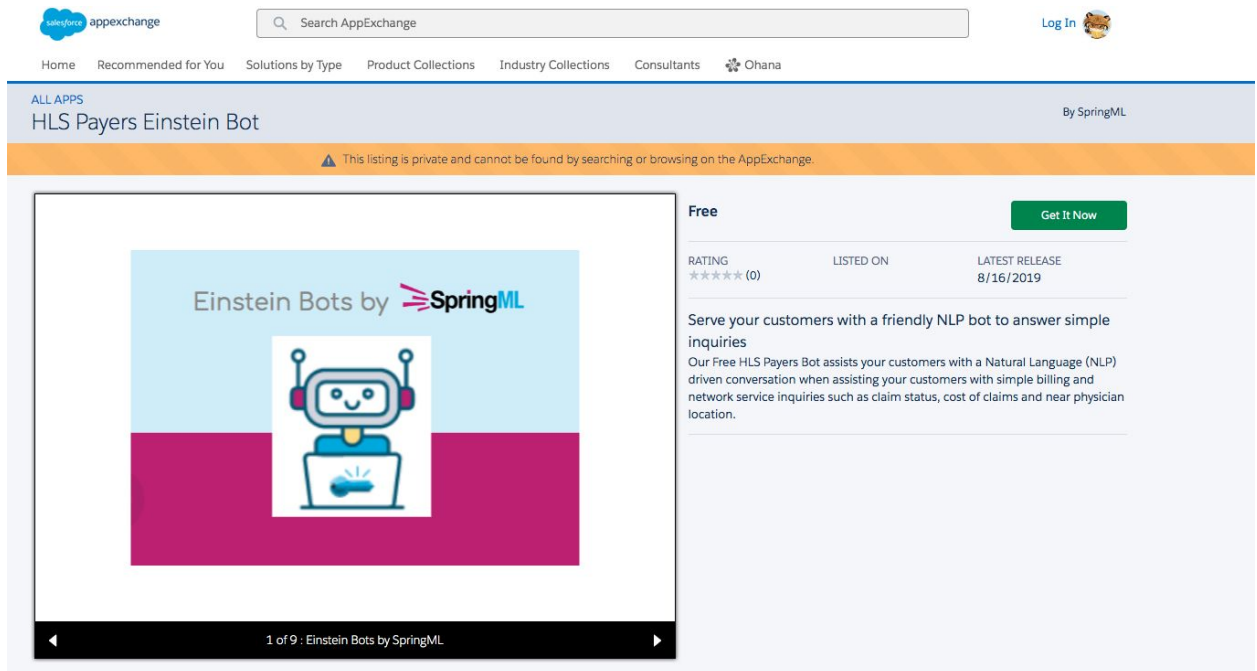
Our Free HLS Payers Bot assists your customers with a Natural Language (NLP) driven conversation when assisting your customers with simple billing and network service inquiries such as claim status, cost of claims and near physician location.

This document provides details on how to install Insurance Einstein Bot and deploy in your org to use this powerful bot for your organization.



Installation the package

- Get the package installation link for “HLS Payers Einstein Bot” from the AppExchange Listing -
 - <https://appexchange.salesforce.com/appxListingDetail?listingId=a0N3A00000FMi5fUAD>
- Or install directly using following link -
 - <https://login.salesforce.com/package/installPackage.apexp?p0=04t3i000001n6PI&isdt=p1>
- Select the **Install for All Users** option. Click **Install**.
- Give it a minute to finish installing and then click **Done!**
- You should see a confirmation page like the following:



Pre-requisites:

- Enable “**Chat (Live Agent)**”, “**Omni-Channel**”.

Documentation Reference Links:

Omni-Channel Setup -

https://help.salesforce.com/articleView?id=console_lex_service_setup_omnichannel.htm&type=5

Chat Setup -

https://help.salesforce.com/articleView?id=live_agent_lightning_experience_setup_flow.htm&type=5

- Add the Configured Presence Statuses to required profiles.

Documentation Reference Link:

https://help.salesforce.com/articleView?id=service_presence_give_access_to_statuses_through_profiles.htm&type=5

- Enable “**Einstein Bots**” and connect it to the **Live Agent Chat** channel.
- Register with **Einstein Platform** for **Einstein Intent Training**.

- Create a Self-Signed Certificate and register your bot with Einstein Platform Services for Artificial Intelligence.

Configurations to use the bot

1. Permission your Bot

- Navigate to Lightning Setup and use the Quick Find box in the upper left corner of the screen to search for **Permission Sets**. Click on the Permission Set named **sfdc.chatbot.service.permset**
- Bot will need access to the following Objects and Fields:
 - **Accounts**
 - Object Permissions – Read, View All
 - Add the below mentioned fields to required account page layouts
 - Field Permissions –
 - **First Name(hlspayerbot__First_Name__c)**(Edit)
 - **Last Name (hlspayerbot__Last_Name__c)**(Edit)
 - **ZipCode(hlspayerbot__ZipCode__c)**(Edit)
 - **MemberId(hlspayerbot__MemberId__c)**(Edit)
 - **PlanId(hlspayerbot__PlanId__c)**(Edit)
 - **Contacts**
 - Object Permissions - **Read,Create, View All**
 - Field Permissions -
 - **Name (Edit)**
 - **Email (Edit)**
 - **MembershipPlans**
 - Object Permissions - **Read, View All**
 - Field Permissions -
 - **Name(hlspayerbot__Name__c) (Edit)**
 - **ExpiryDate(hlspayerbot__ExpiryDate__c) (Edit)**
 - **Type(hlspayerbot__Type__c) (Edit)**
 - **PercentCoverage(hlspayerbot__PercentCoverage__c) (Edit)**
 - Read Permission for all auto number fields
 - **MembershipTransactions**
 - Object Permissions - **Read, View All**
 - Field Permissions -
 - **Claim No(hlspayerbot__Claim_No__c) (Edit)**
 - **Claim Status(hlspayerbot__Claim_Status__c) (Edit)**

- **Claim Status**
Reason(hlspayerbot__Claim_Status_Reason__c) (Edit)
- **Member Payment**
Status(hlspayerbot__Member_Payment_Status__c) (Edit)
- **MemberId(hlspayerbot__MemberId__c) (Edit)**
- **OutofthePocket(Not Covered)(hlspayerbot__OutofthePocket_Not_Covered__c) (Edit)**
- **PaymentDate(hlspayerbot__PaymentDate__c) (Edit)**
- **PhysicianId(hlspayerbot__PhysicianId__c) (Edit)**
- **PlanId(hlspayerbot__PlanId__c) (Edit)**
- **Service Code(hlspayerbot__Service_Code__c) (Edit)**
- **Service Type(hlspayerbot__Service_Type__c) (Edit)**
- **ServiceName(hlspayerbot__ServiceName__c) (Edit)**
- **TotalAmount(hlspayerbot__TotalAmount__c) (Edit)**
- **TransactionId(Name) (Read)**

- **Physicians**

- Object Permissions - **Read, View All**

- Field Permissions -

- **City(hlspayerbot__City__c) (Edit)**
 - **First Name(hlspayerbot__First_Name__c) (Edit)**
 - **Last Name(hlspayerbot__Last_Name__c) (Edit)**
 - **Network(hlspayerbot__Network__c) (Edit)**
 - **PhysicianId(Name) (Read)**
 - **Service Type(hlspayerbot__Service_Type__c) (Edit)**
 - **State(hlspayerbot__State__c) (Edit)**
 - **ZipCode(hlspayerbot__ZipCode__c) (Edit)**

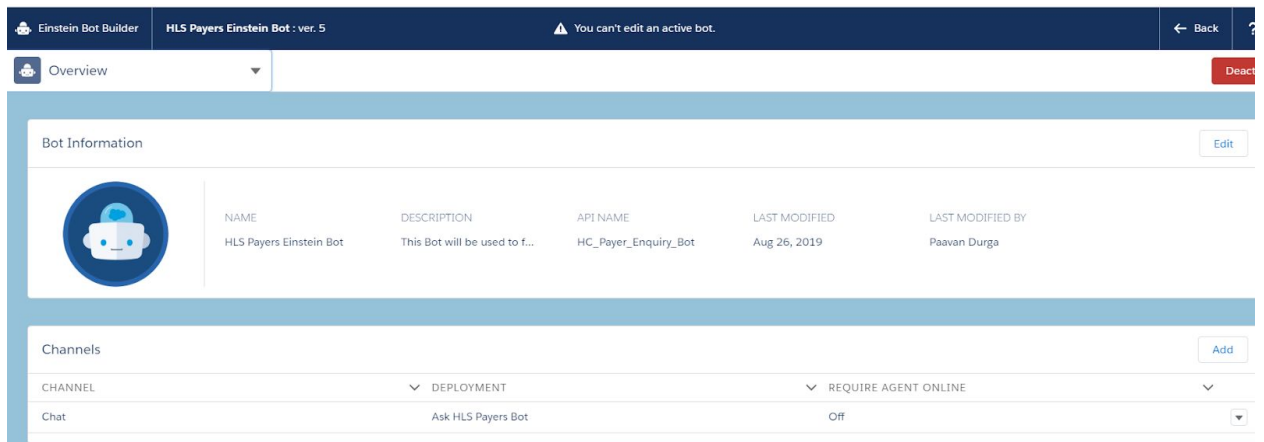
- Provide Access to **hlspayerbot.*** related Apex classes so that Bot works as intended and they would be available through the installed package.

Please note that the above step is mandatory otherwise Bot will not give desired information.

- **All set!** Your bot can now access a little bit of customer data and some example platform Actions!

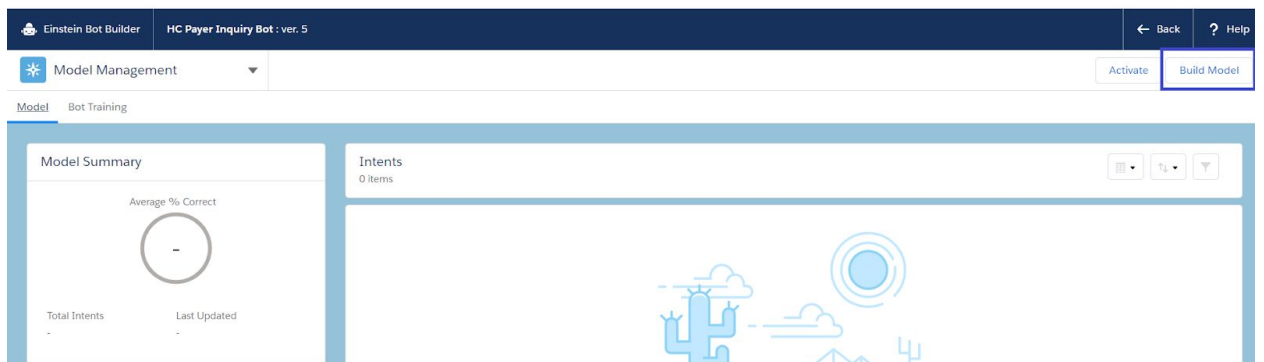
2. Add your Chat To Bot

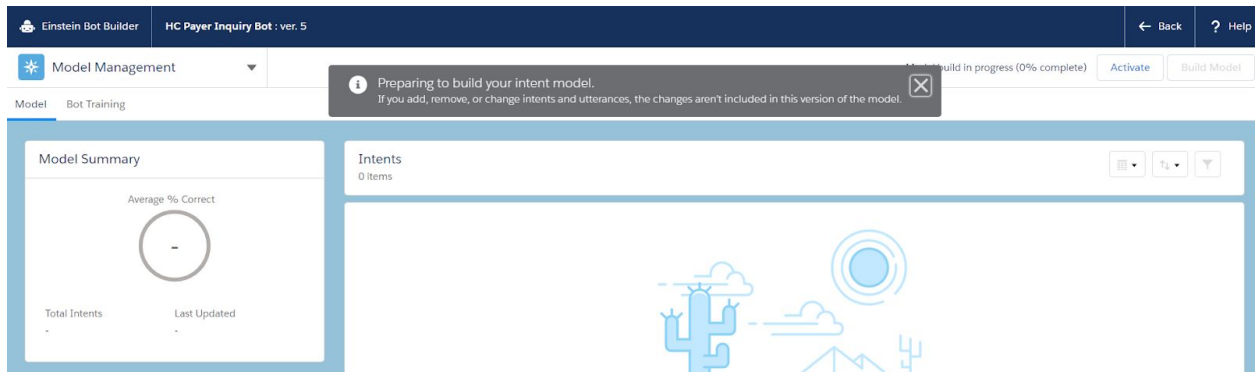
- From the **Bot Builder** menu, click **Overview**.
- In the Channels section, click **Add**.
- Select the channel and deployment for your bot.
- To disable the chat button when no agents are available for escalation, select **Require Agent Online**.
- Please find below screenshot for the same.



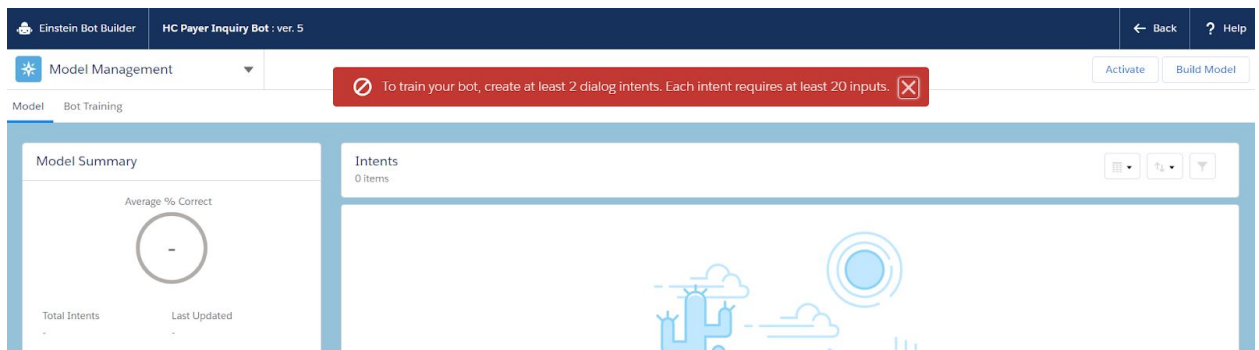
3. Train and Activate your bot

- From the **Bot Builder** menu, click **Model Management**.
- In the Model Management screen, click **Build Model** button then you should see progress for Build Training , find below screenshot for the reference.





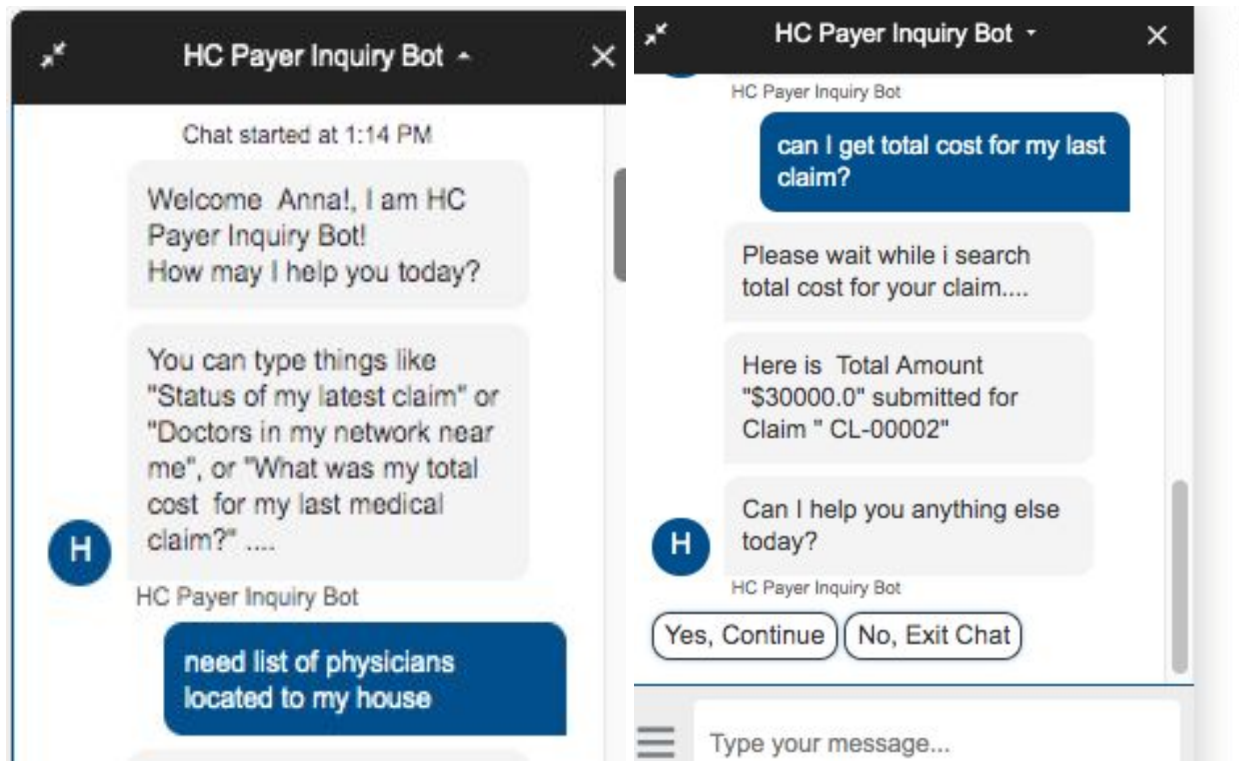
- If you get below error then remove the Utterances From **‘Welcome’**, **‘Confused’**, **‘End Chat’**, **‘Transfer to Agent’** dialogs and Turn Off Einstein in the above mentioned dialogs.



- Once training is completed, click Activate button.

USING PREVIEW

1. Navigate to Lightning Setup > Einstein Bots > HC Inquiry Bot.
2. Navigate to the Dialogs tab in Einstein Bot Builder. Make sure your bot is Active and click on the Preview button to chat with your bot
3. Your Snap-in should say “Chat with an Expert”.
4. Click the Start Chatting button! You should now see your bot display the welcome message and your clickable navigation menu:
5. Bot is ready for you to test.



Using Bot in Communities

1. Ensure steps mentioned below on “**How to configure bot for communities**” section is done.
2. On right side click on the Chat button, calling it “**HLS Einstein Bot**” as an example.
3. Bot is ready for you to test.

How to configure bot for communities

- In setup navigate to “**All Communities**” and open the builder.
- From the lightning components menu drag and drop “**Embedded Service (Snap-ins Chat)**”
- Then set the Chat Deployment to “**Chat**” (or whatever is your Chat Deployment)
- Publish your community.

Please reach out to support@springml.com for any queries regarding the Bot.

We would love to learn more about your Einstein projects and explore ways in which we can help you.

Einstein Bots by  **SpringML**

