**Creating Chabot in Salesforce using chatGPT**

**Chatbots** have become an integral part of customer service and sales operations. They are used to automate customer service tasks, provide personalized customer experiences, and even help close sales.

**Salesforce** is one of the most popular customer relationship management (CRM) platforms, and it is no surprise that many companies are looking to integrate their chatbot with Salesforce. By integrating a chatbot with Salesforce, companies can leverage the power of both platforms to create a more efficient and effective customer service and sales process.

**ChatGPT** is a powerful chatbot platform that can be easily integrated with Salesforce. ChatGPT provides a comprehensive set of features that allow companies to create powerful chatbots that can be used to automate customer service tasks, provide personalized customer experiences, and even help close sales.

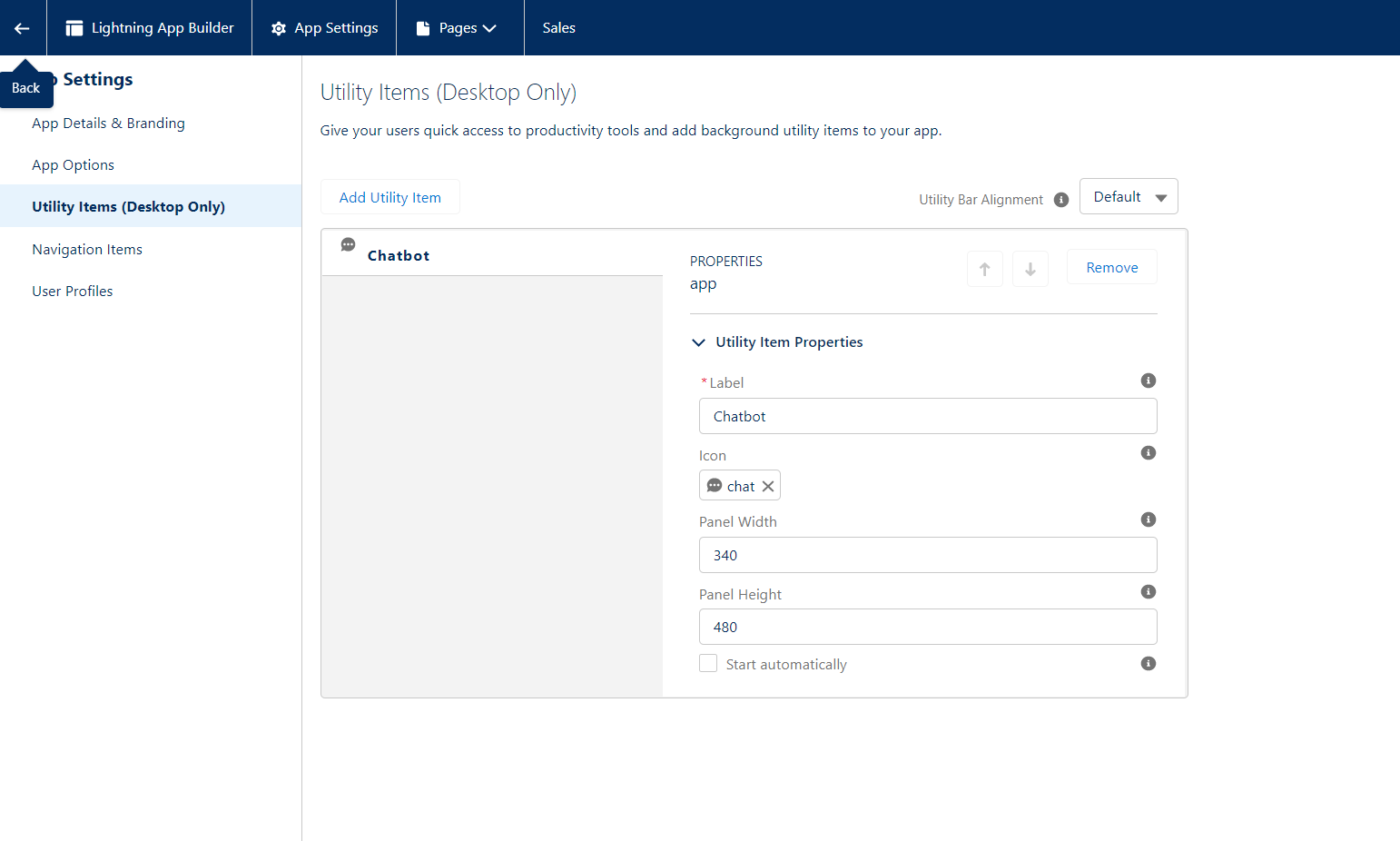
Creating a chatbot using ChatGPT is a great way to provide customers with quick and efficient service. It can help you automate mundane tasks and provide customers with personalized recommendations. With the help of Salesforce, you can easily integrate your chatbot with your existing customer service and sales strategies.  
  
**Pre-req:**

* Salesforce account ( <https://developer.salesforce.com/signup> )
* ChatGPT account ( https://chat.apps.openai.com/auth/login )

**Steps:**

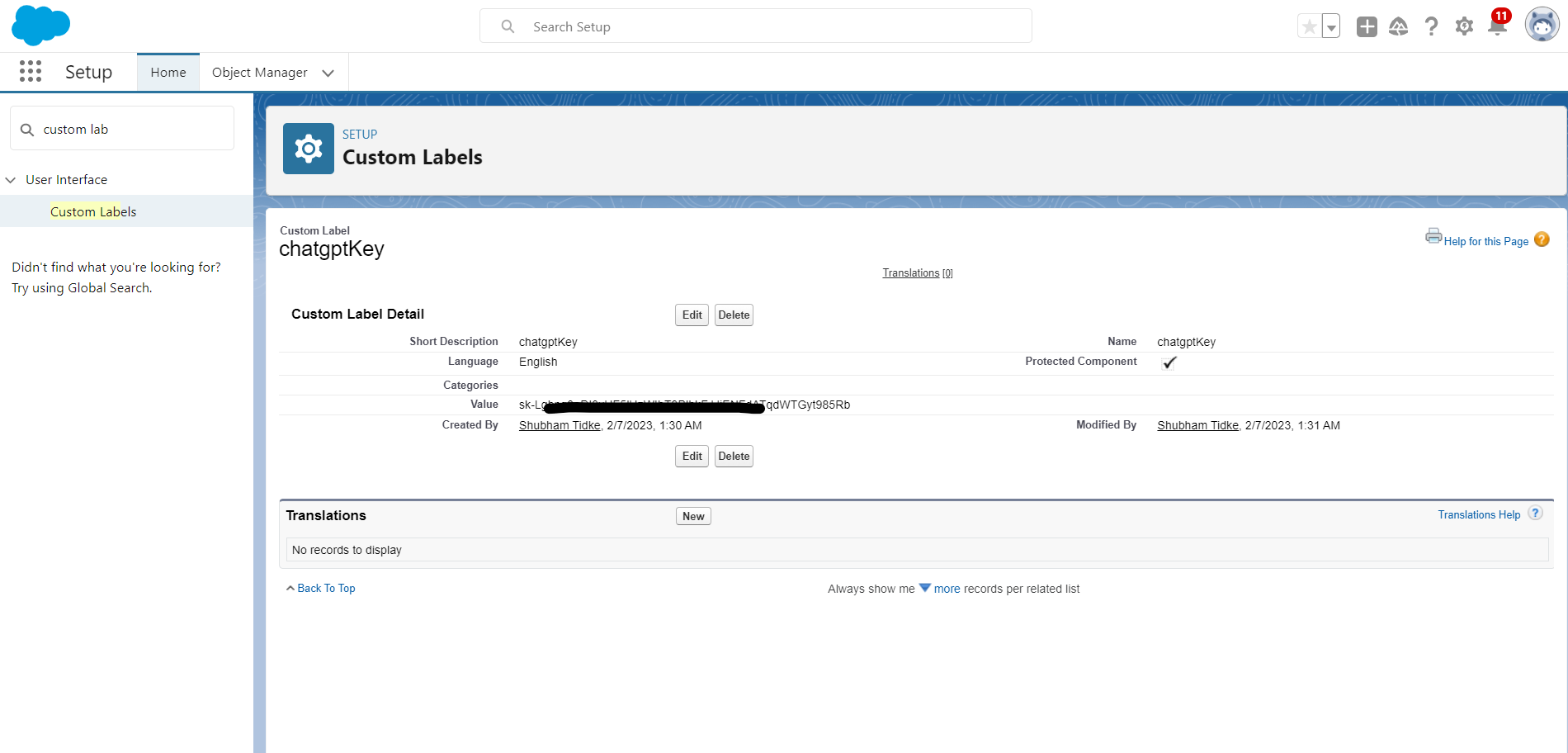
1. Add chatbot as utility item in your salesforce app

In your salesforce org, setup -> App Manager -> Utility Items -> Add Utility item,   
and search chatbot . click on it and save.  
if you could not find it,skip the step and follow from step 2,



1. Get Access Key for chatGPT API

To access chatGPT endpoint,we need api key which is unique for every user.  
API Key can be generated using the link : <https://platform.openai.com/account/api-keys>

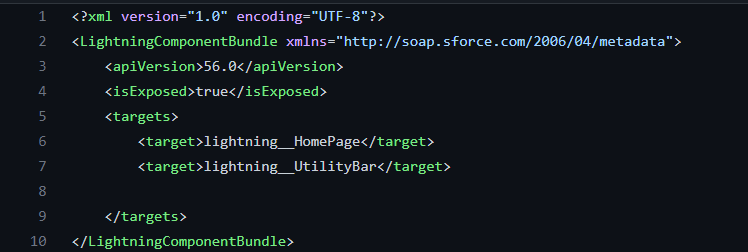
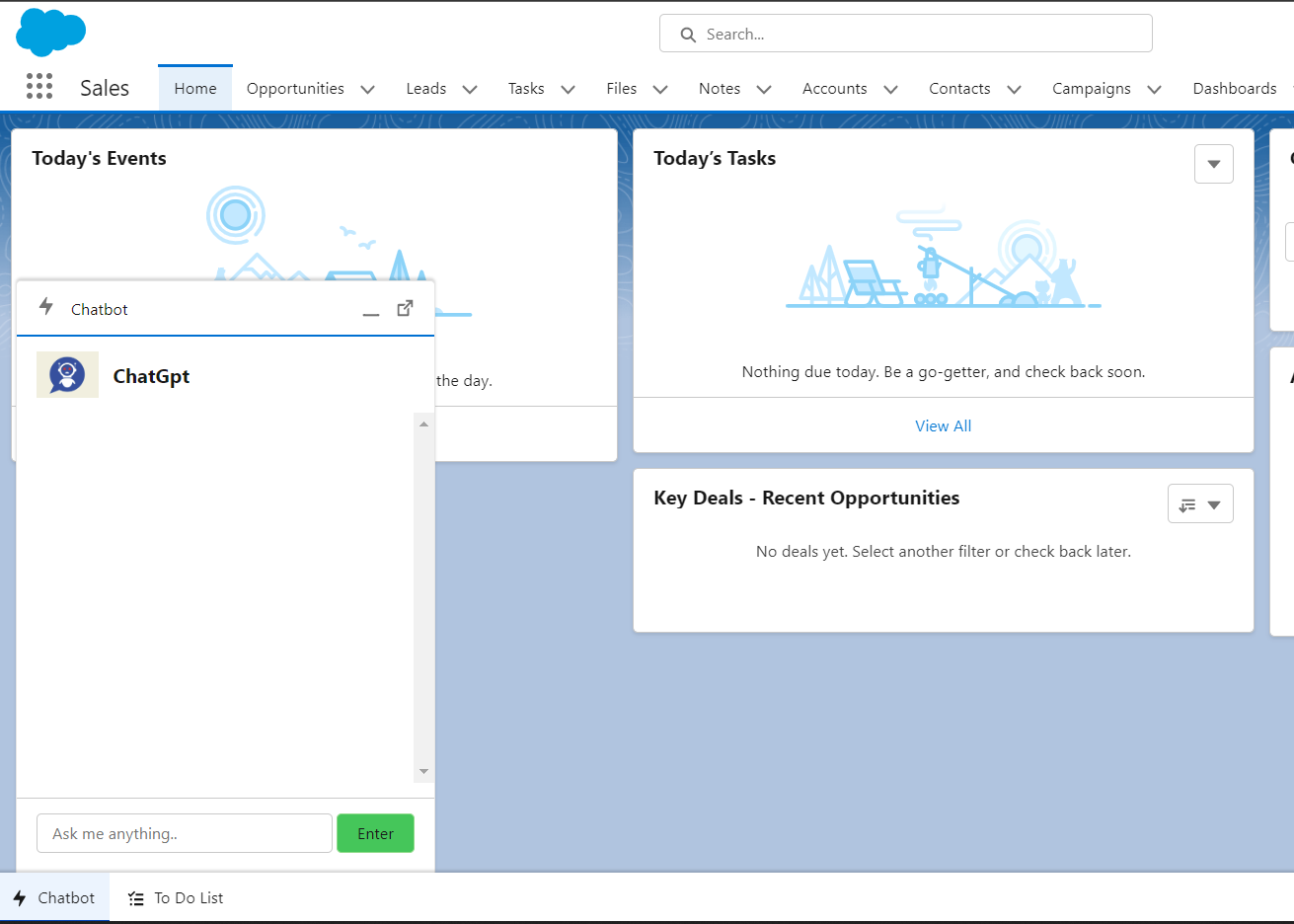
1. Add API Key to Salesforce  
   Add the API Key as custom label hey in your Salesforce org .  
   go to Setup -> Custom Labels -> New Custom Label, and provide any name with API KEY as value.  
     
     
   

The idea behind keeping API key as custom label is to keep your API key hidden. We can easily import this API key in our lightning web component without exposing it publicly.

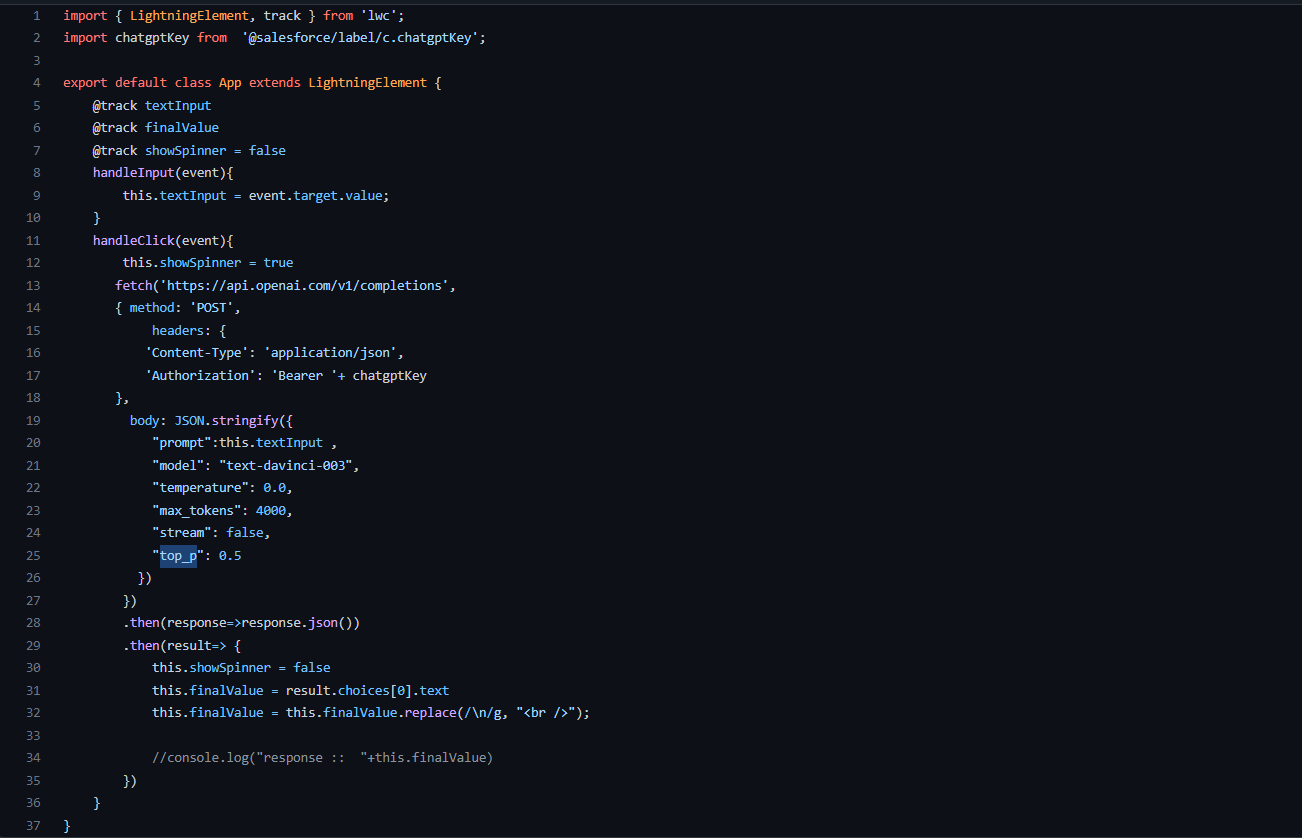
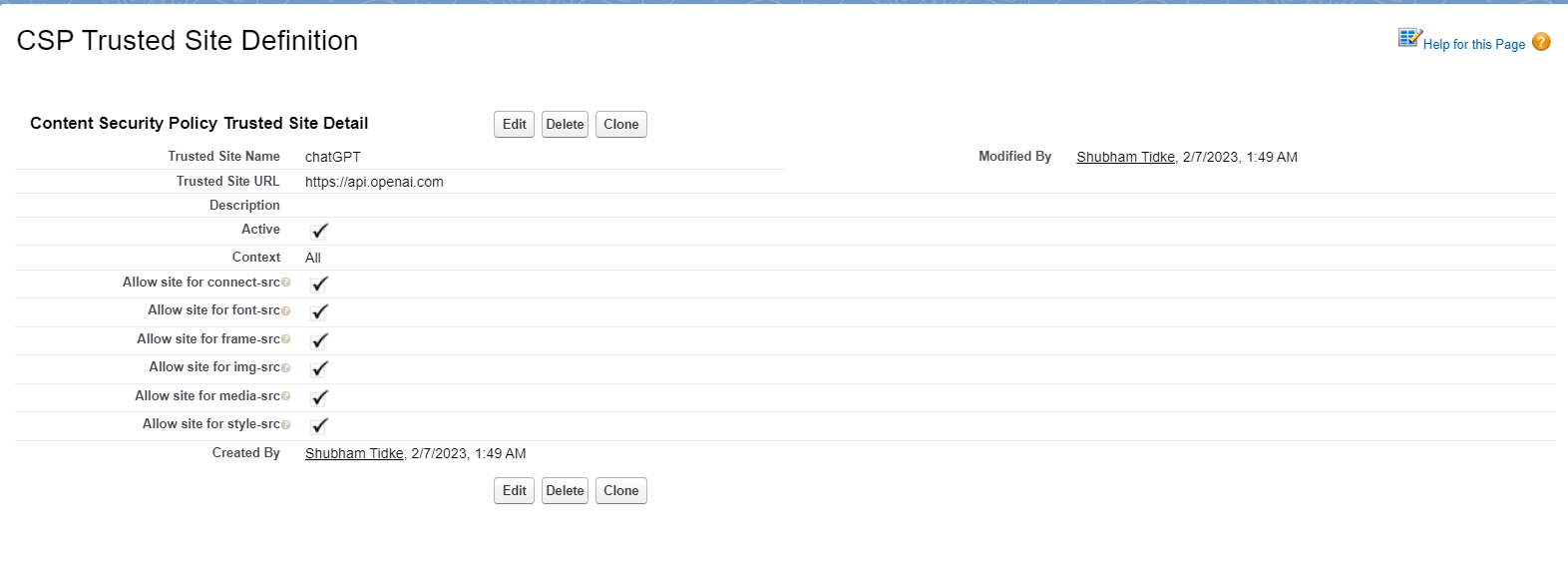
1. Create Lightning Web Component to customize chatbot

Using VS Code, create LWC component. We will be using [completions](https://platform.openai.com/docs/api-reference/completions) endpoint which will answer the questions provided to the chat bot as an input.  
API documentation link : <https://platform.openai.com/docs/models/content-filter>

Create an input field to take input from user in the chat bot with a button on click of which we can load our results. A spinner is also added in HTML, which will load on screen till the API returns with result data.

  
  
Add the targets in meta.xml file of your component and deploy the code.  
  
  
  
In order to deploy it in utility bar, we need to add <target>lightning\_\_UtilityBar</target>   
When you deploy the code, go to setup -> App Manager -> Utility Items -> Add Utility item  
and in the search box find your custom component’s name. Add the component and save.  
It will look something like this,  
  


In JS, import the custom label which we created the org to store api key. Create a variable to store text input received from user via chatbot and in theevent-type of button on the chatbot we will send a POST request to chatgpt endpoint.  
There are few headers which we need to pass to the endpoint,  
1. Prompt : it will be the input text (question we will be asking to the chatbot)  
  
2. Model : OPEN API supports differentmodels which has different capabilities,here we will use  
“text-davinci-003” which understands natural language and generate longer inputs (4000 tokens)  
  
3. Tempreature : Using this header we can customize the creativity of the answer.Lower the temperature more straight-forward the answers.  
  
4. max\_tokens : max limit of pieces of words we will be using. 1000 tokens are around 750 words.  
  
5. Stream: to stream back partial progress. If set, tokens will be sent as data-only server-sent events as they become available, with the stream terminated by a data: [DONE] message.  
  
6. top\_p : An alternative to sampling with temperature, called nucleus sampling, where the model considers the results of the tokens with top\_p probability mass. So 0.1 means only the tokens comprising the top 10% probability mass are considered.

Using fetch we will send request to endpoint,  
  
  
After deploying, add the endpoint as CSP trusted sites in your org  
  
  
Once done, you are go to run your chatbot supported by ChatGPT. Below are some snippets,  
You can get the full code [here](https://github.com/Shubham-Tidke/Salesforce-Practice/tree/Master/LWC/LWC-Applications/ChatGPTIntegration).

