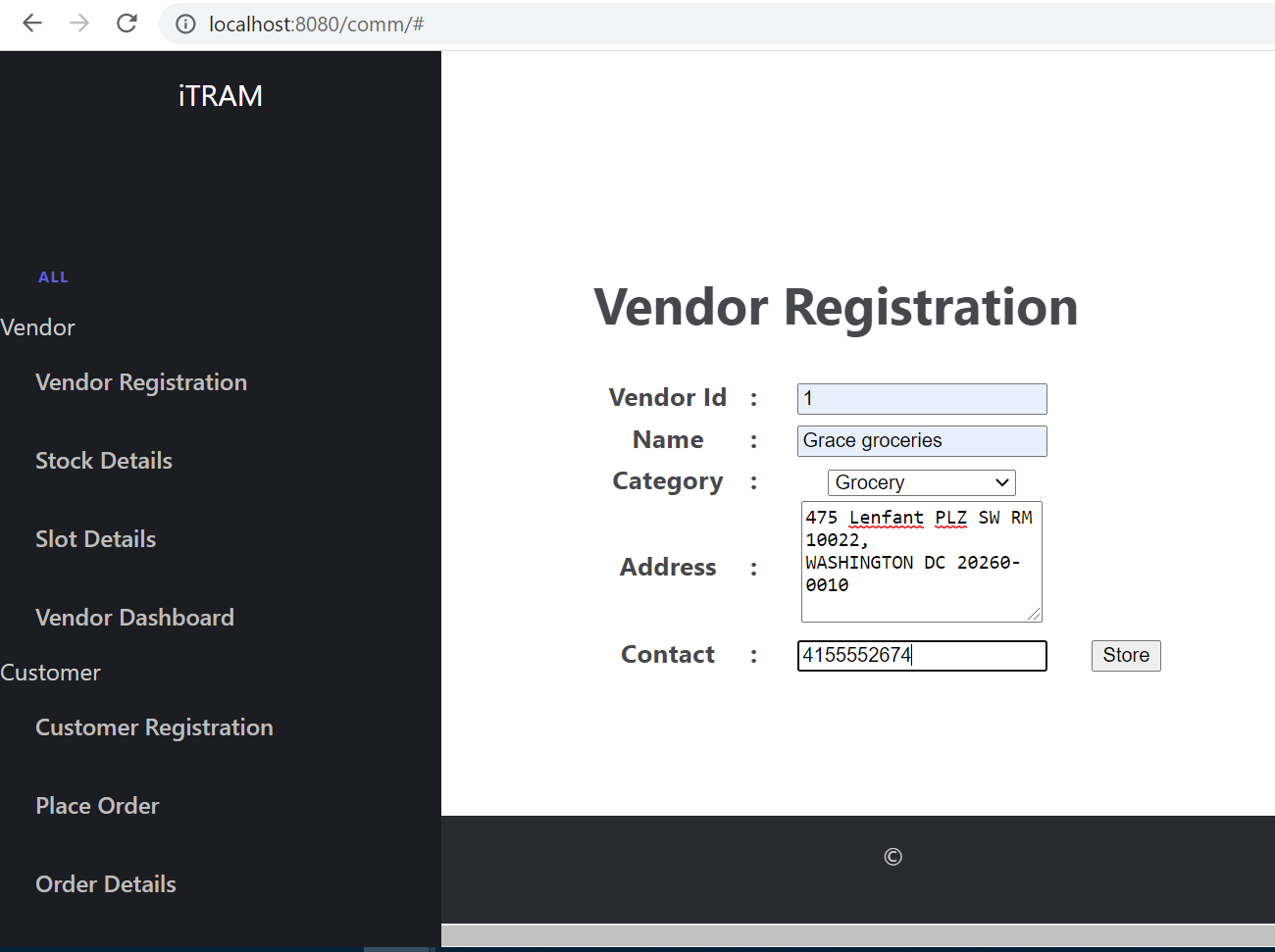
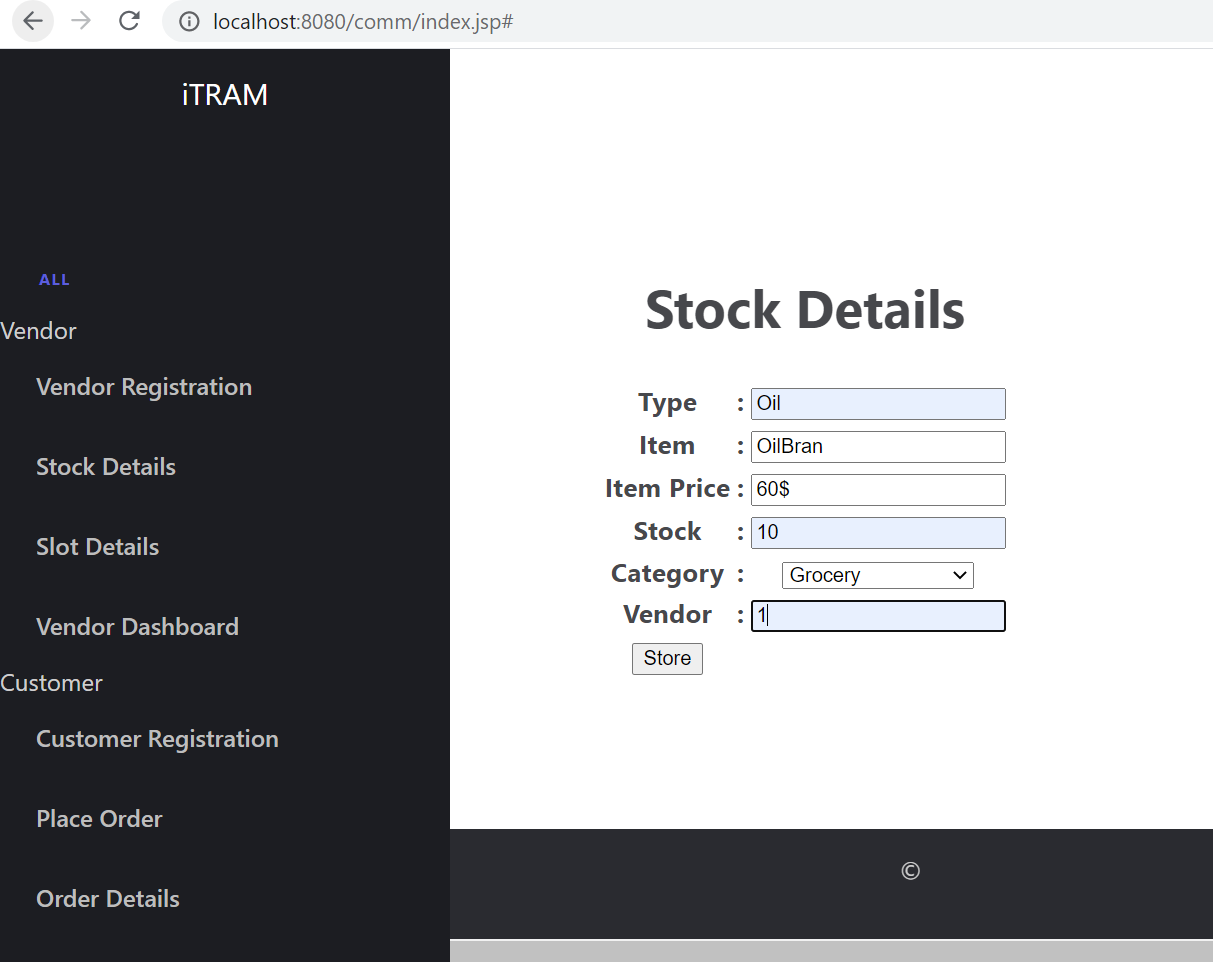
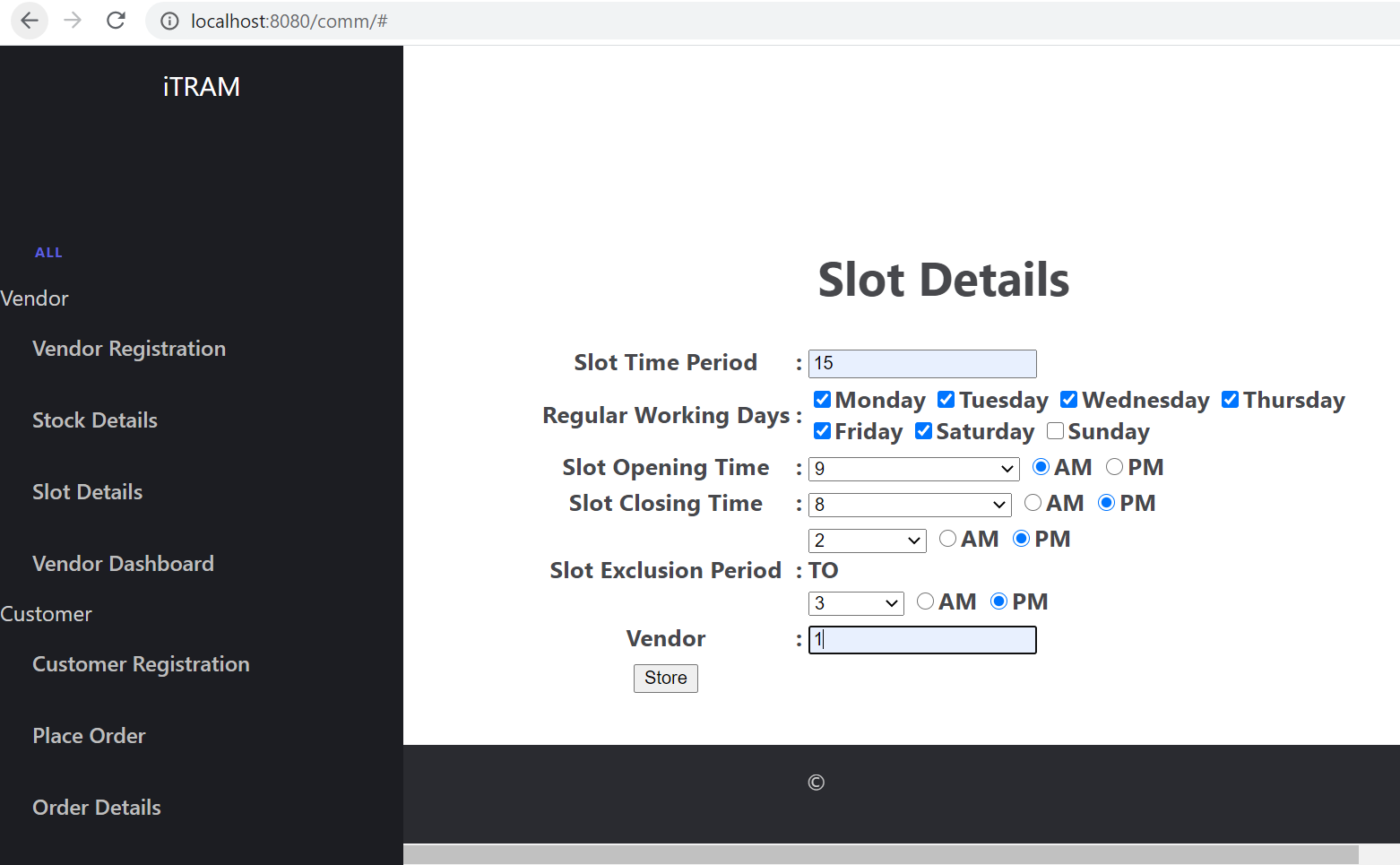
**Web application for vendor and customer registration**

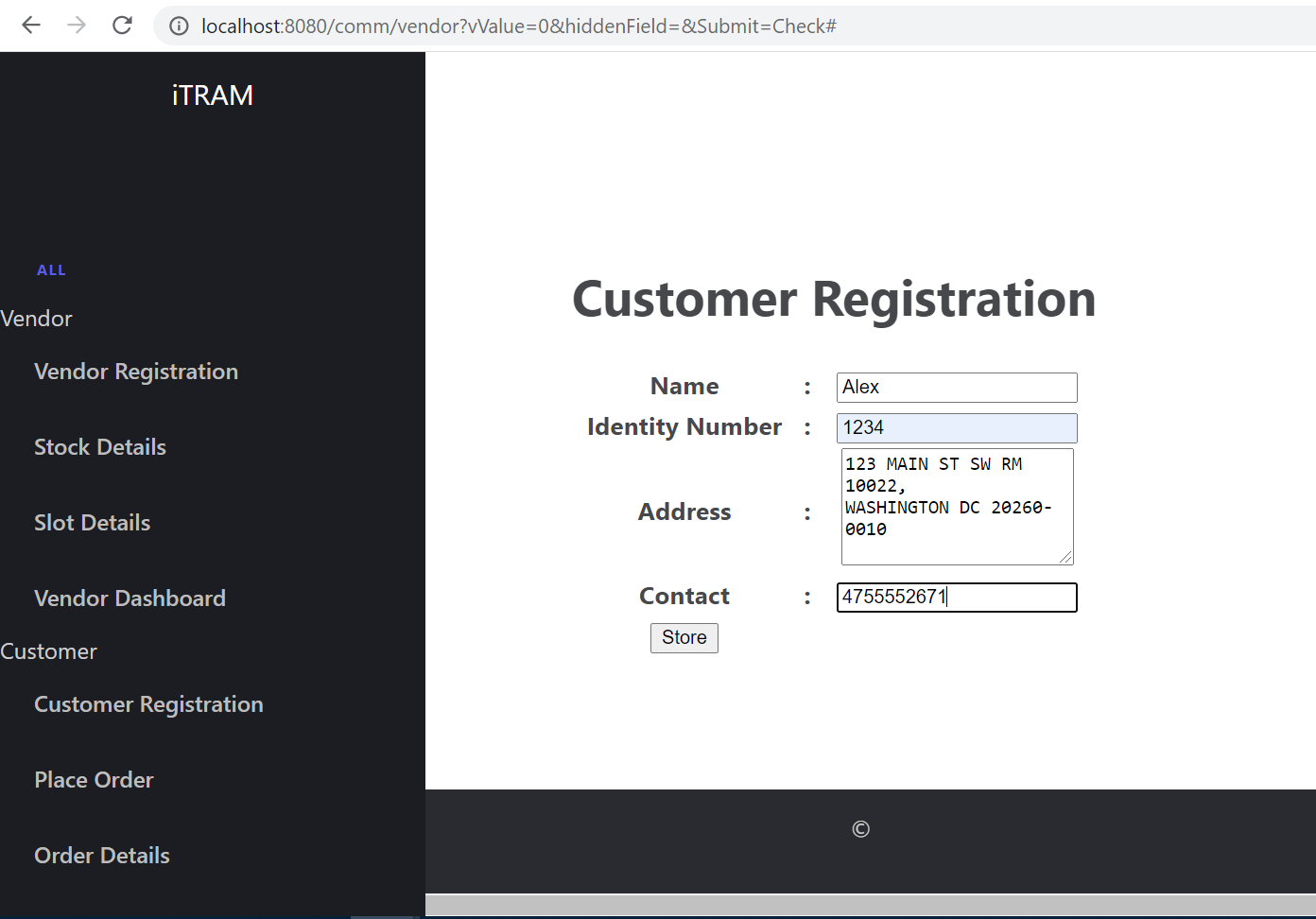
Vendor registers the stock and slot availabilities in the app along-with the location.

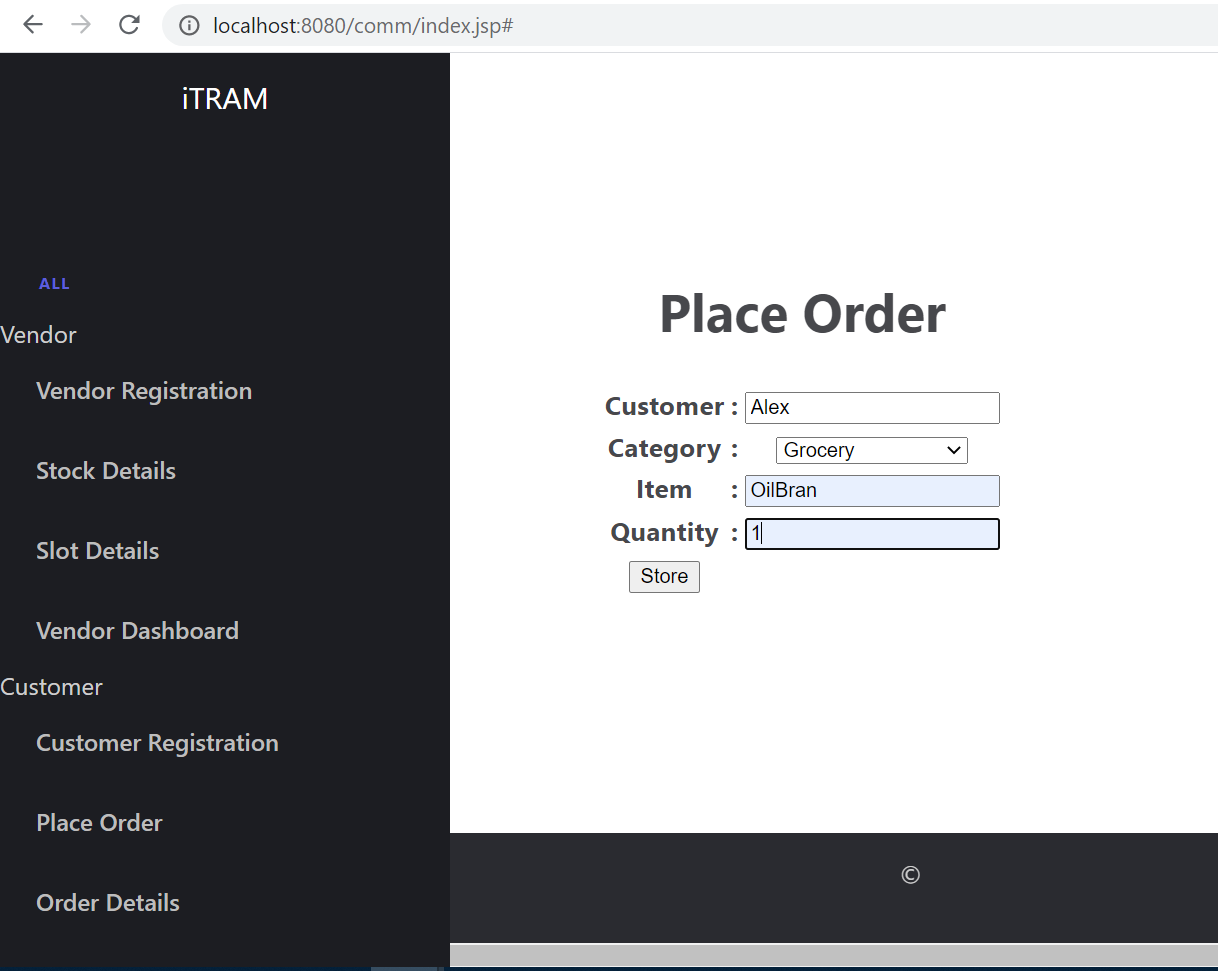




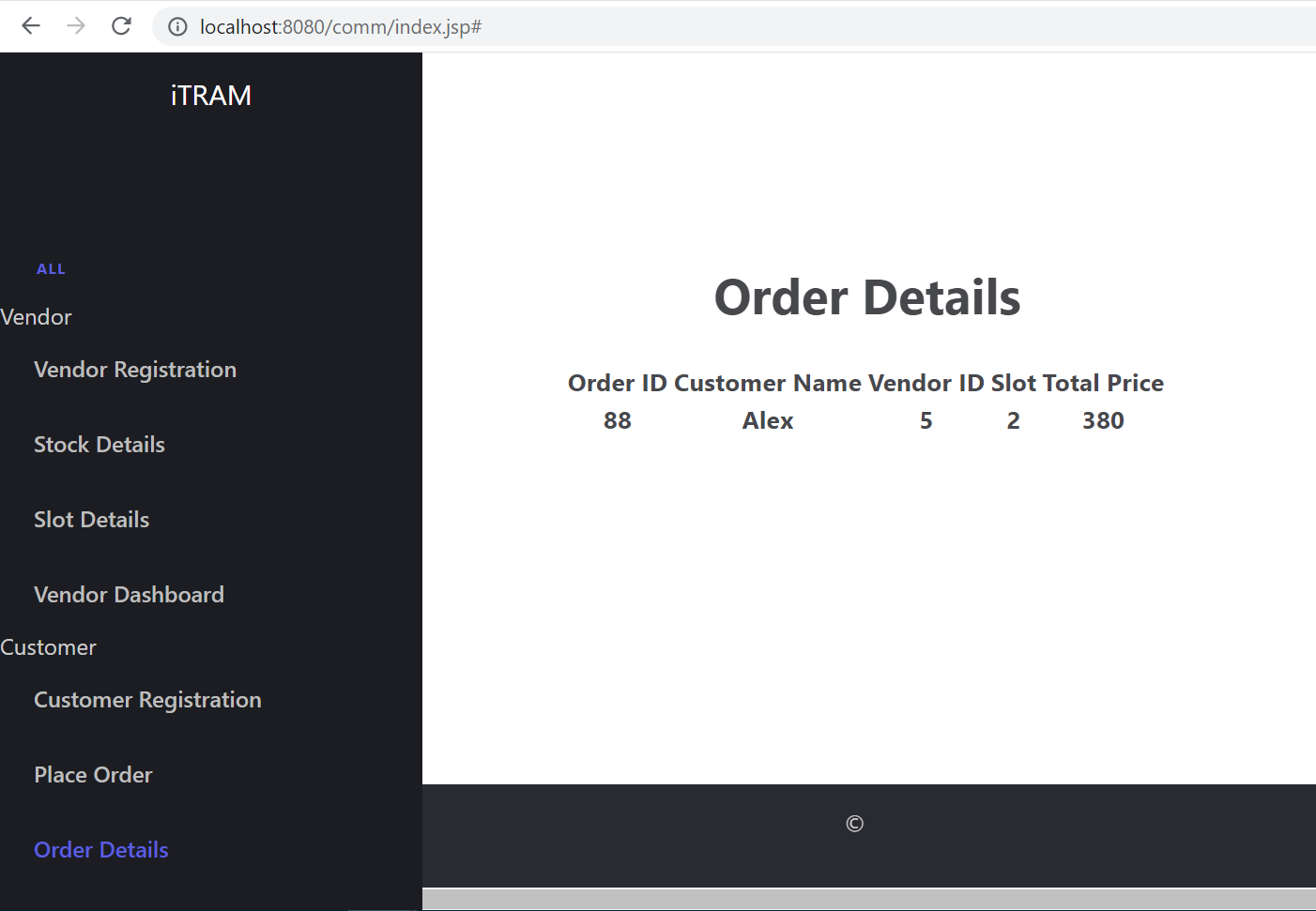


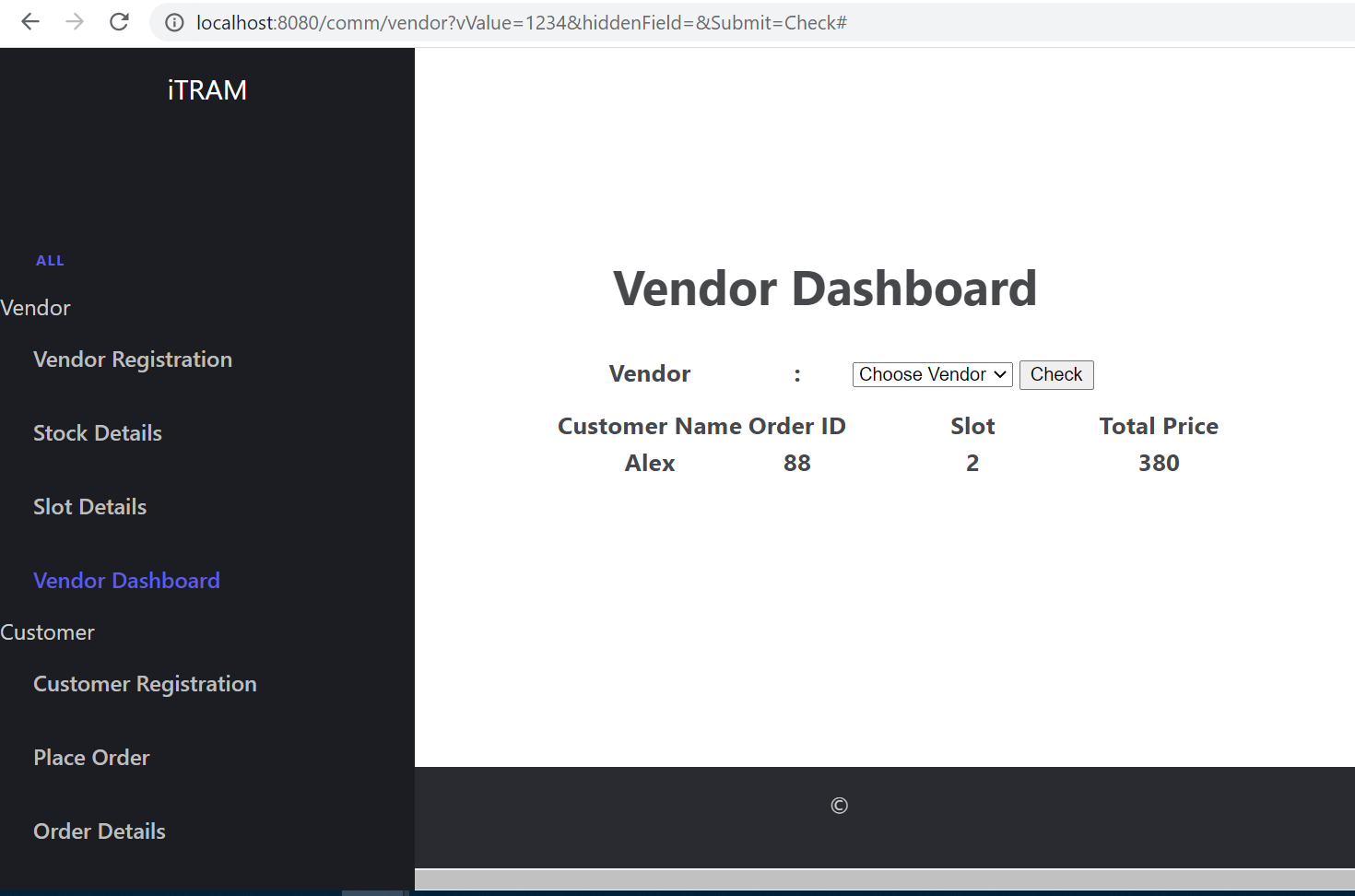
Customer regsiters and places an order in the app for the inventory needed.



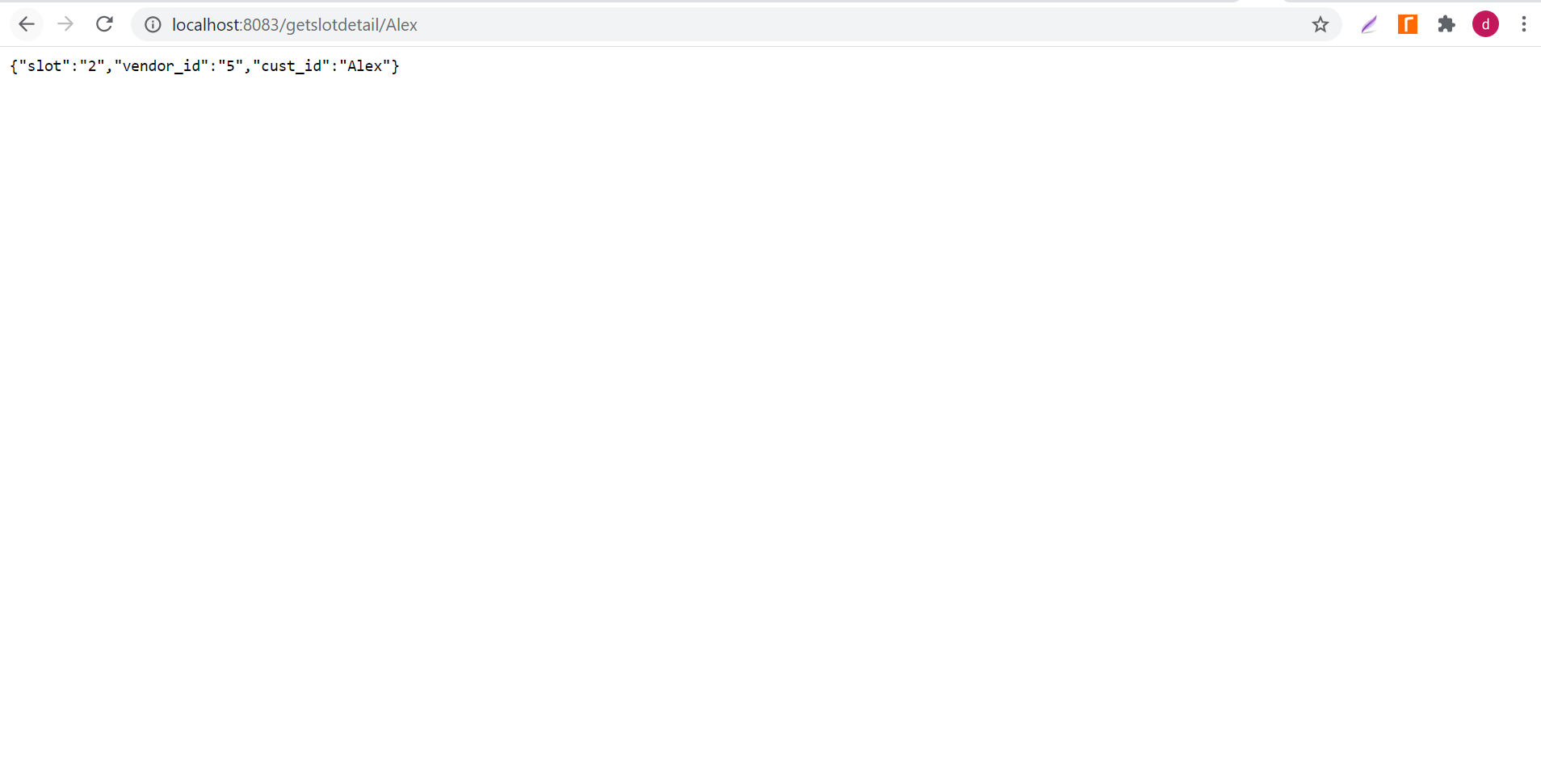


Customer order details are populated by choosing the location of the vendor closer to the customer location and vendor is also able to see the dashboard of the slot availabilities.



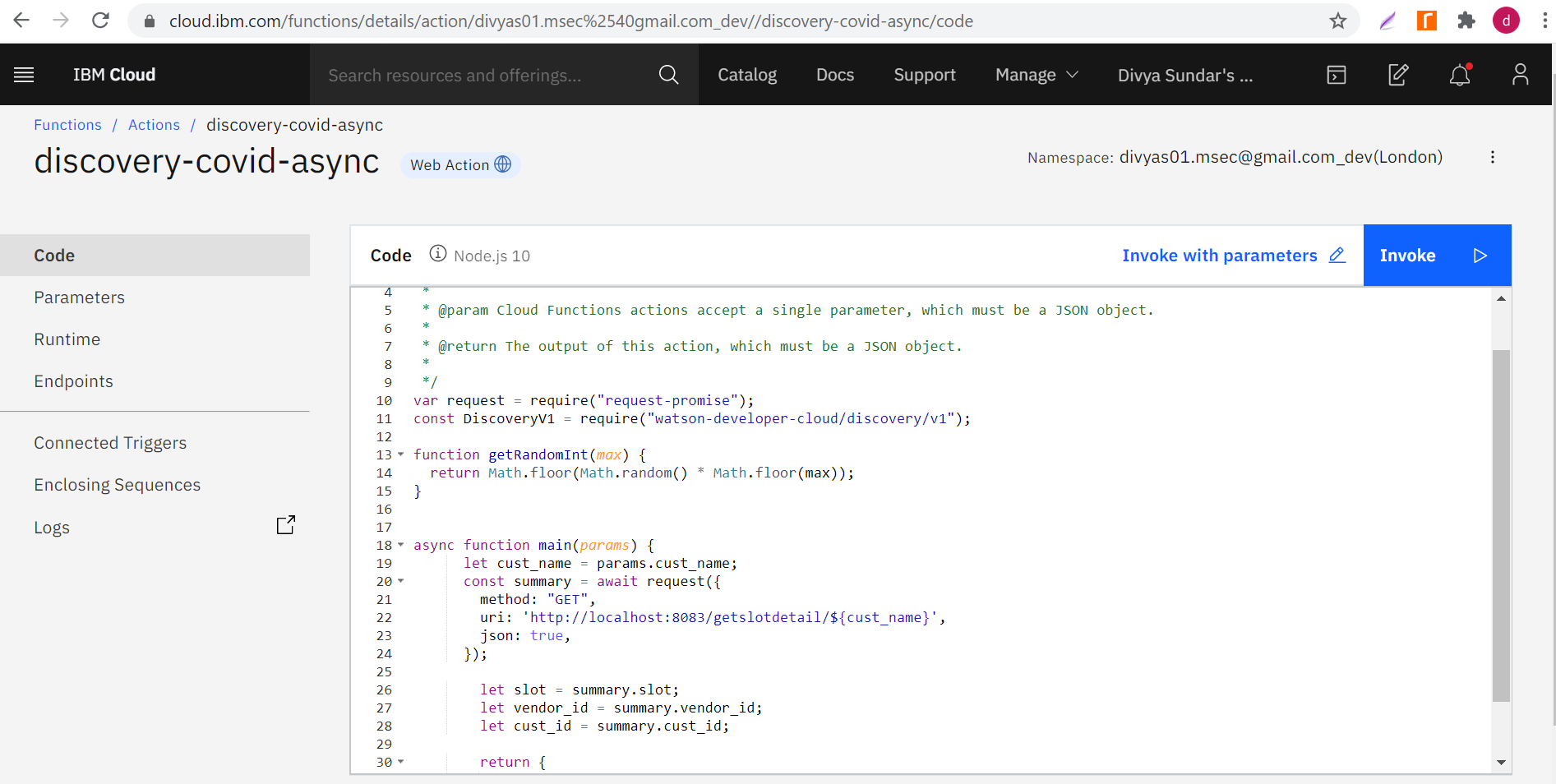


**Rest Response for the spring-boot application to give the customer and slot booked**

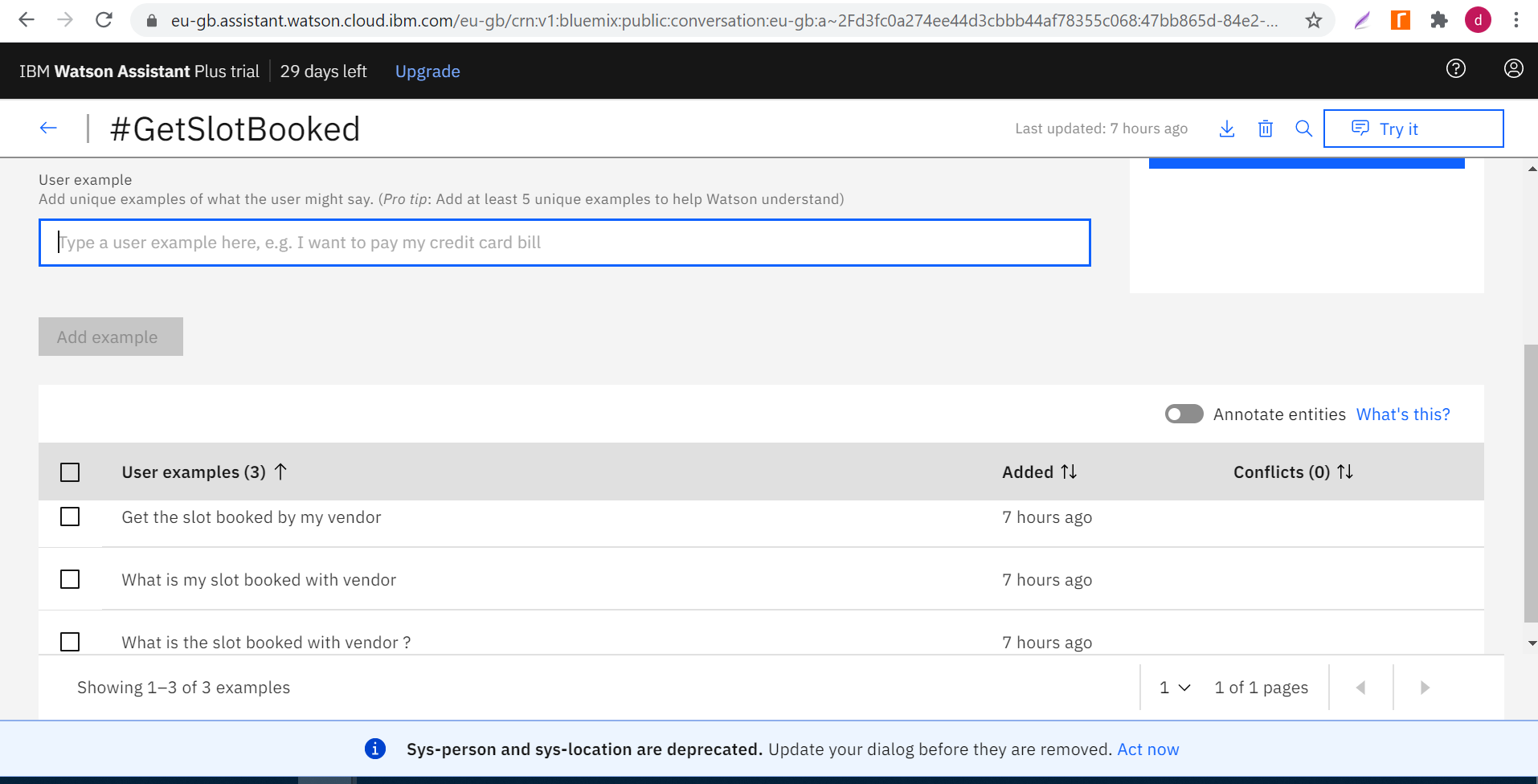


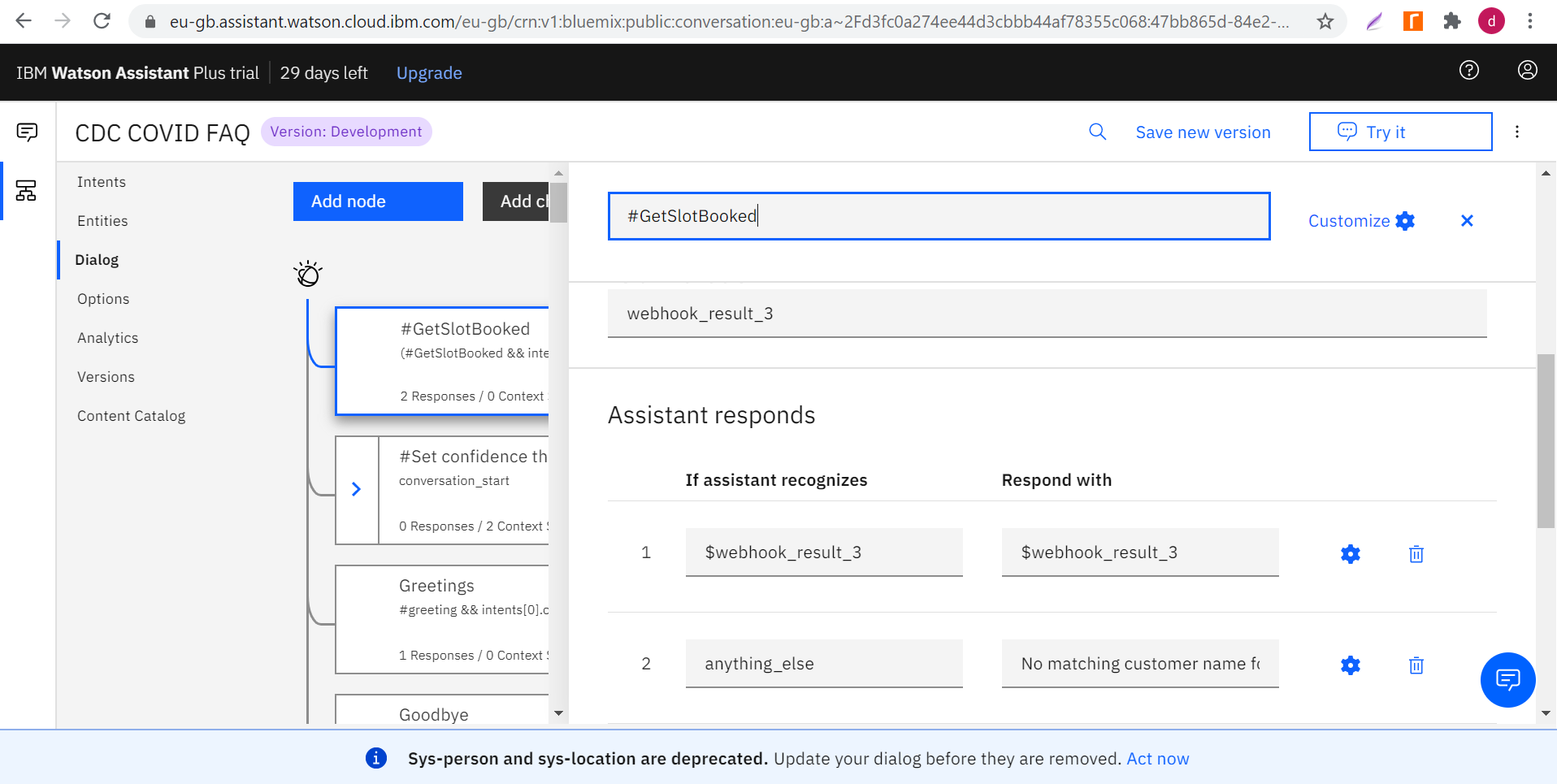
**IBM tools used**

Cloud function having the node 10 code to invoke the external REST call

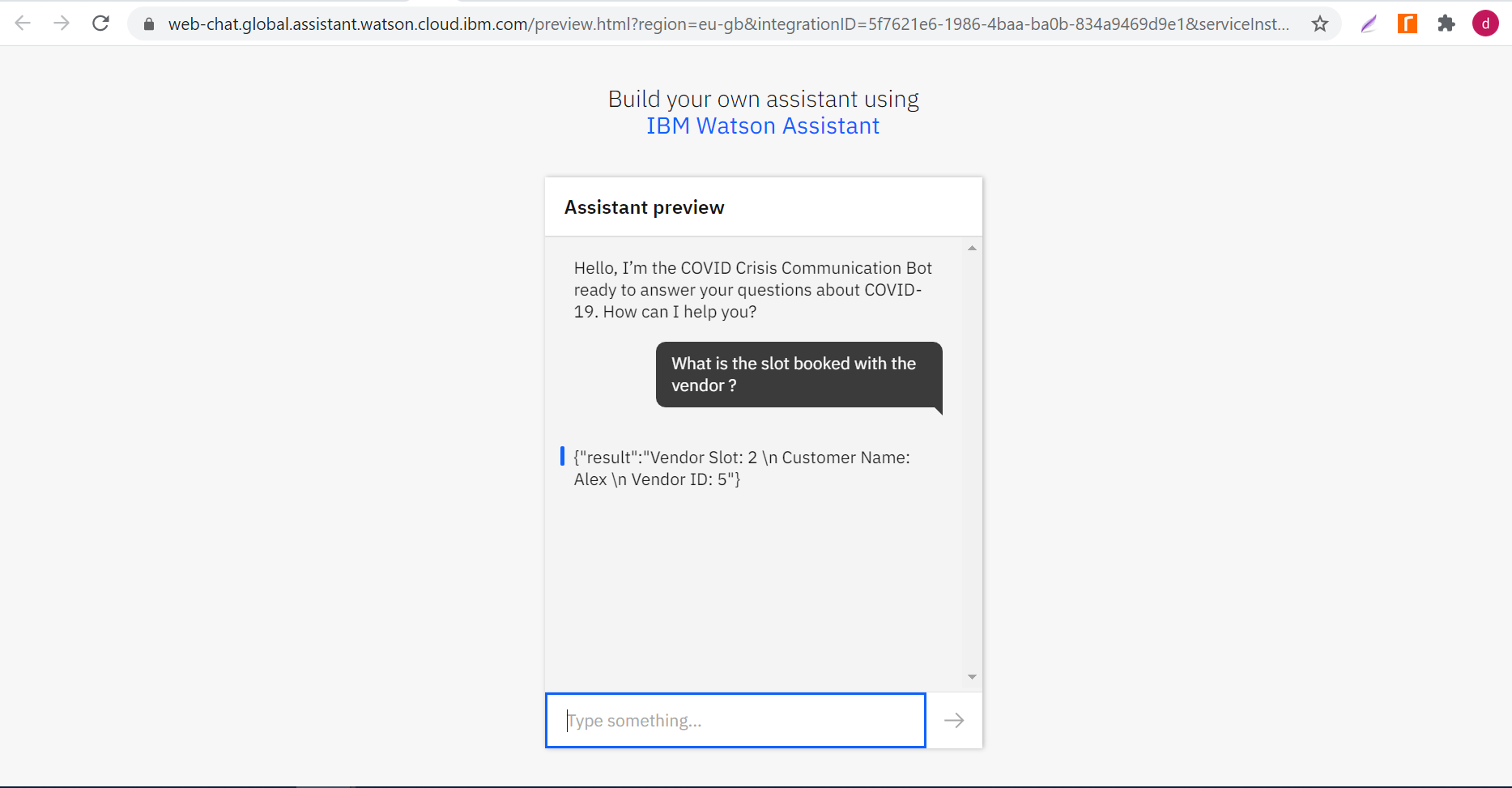


Custom intent (#GetSlotBooked) and dialog skill created to invoke the webhook

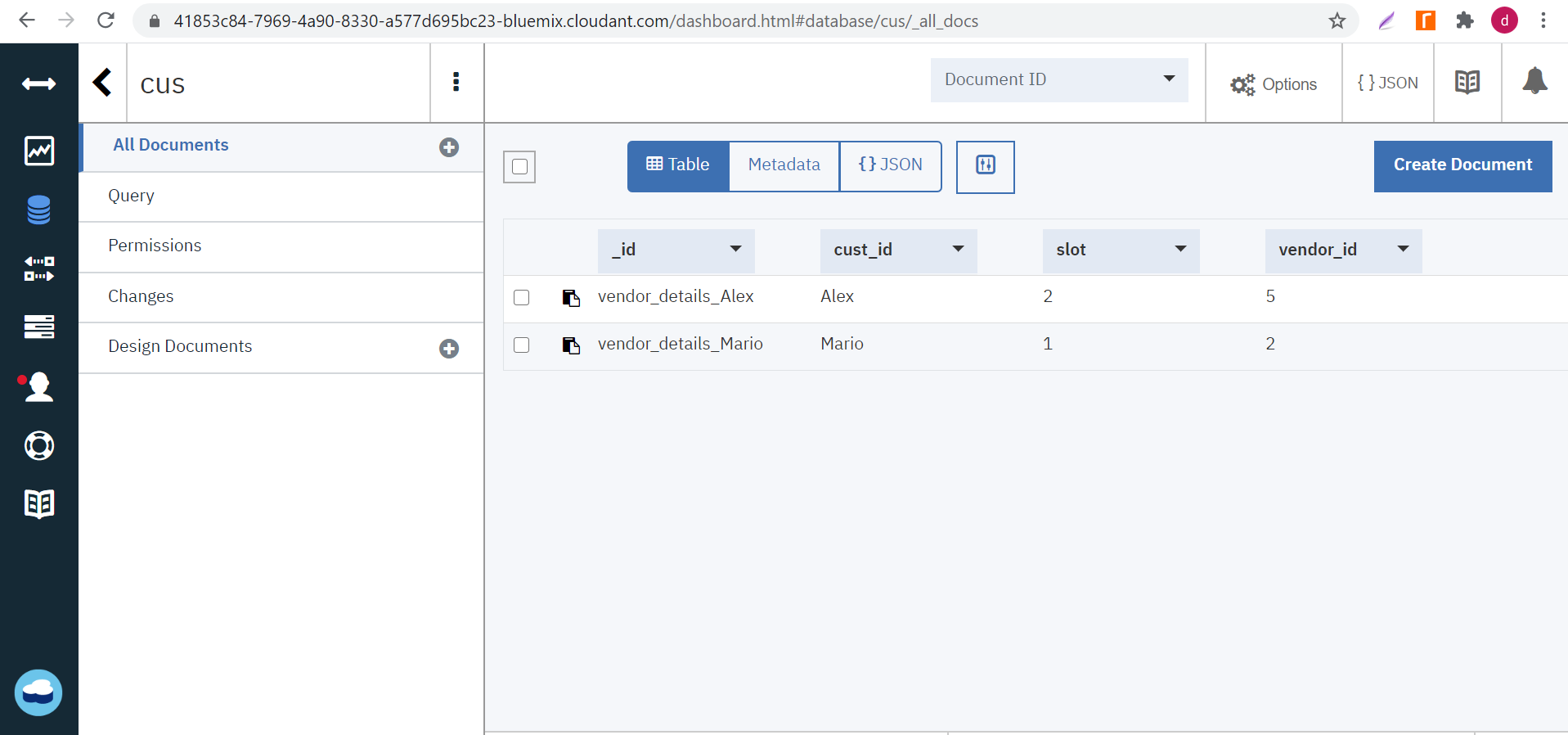




Watson chatbot launched which responds with the customer id and the slot booking id based on the query asked, assuming Alex logs in the chatbot.



Snapshot of cloudant database where the records for vendor slot id, vendor id and the customer id is stored (when the springboot project is started to access the REST calls), which is accessed by the nodejs cloud function that serves a response to the assistant



Snapshots from the local MYSQL database where data is stored in tables for each of the vendor, customer registration and details.

