

# PROJECT DEVELOPMENT PHASE

## SPRINT-4

### CLOUDANT AND SMS

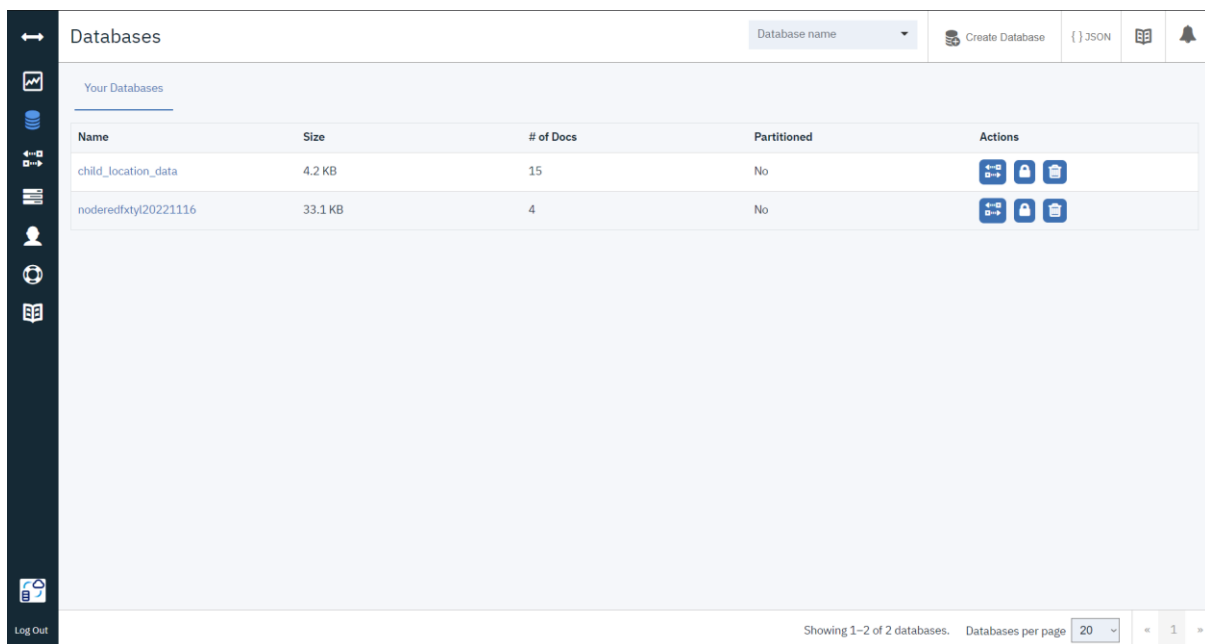
Team ID	PNT2022TMID11653
Project Name	IoT Based Safety Gadget for Child Safety Monitoring and Notification
Maximum Marks	10 marks

#### AIM:







To create cloudant database in IBM cloud and send alert message when the child is out of geofence through fast2sms.

#### CLOUDANT:

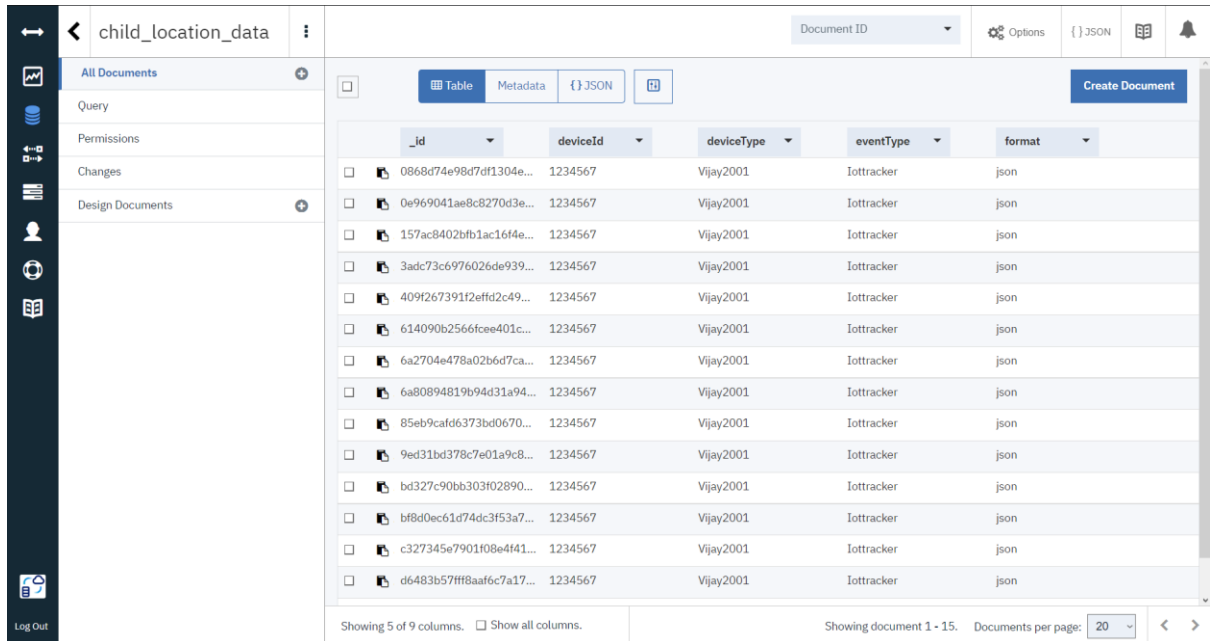
- Created database in IBM cloudant service as child\_location\_data.



The screenshot displays the IBM Cloudant 'Databases' management page. A sidebar on the left contains navigation icons for various cloud services. The main content area shows a table of databases. The table has columns for Name, Size, # of Docs, Partitioned, and Actions. Two databases are listed: 'child\_location\_data' (4.2 KB, 15 docs) and 'noderedfxtyl20221116' (33.1 KB, 4 docs). Each database row has three action icons: a plus sign, a lock, and a trash can. At the top right, there are buttons for 'Create Database', a JSON icon, and a bell icon. At the bottom right, a pagination bar indicates 'Showing 1-2 of 2 databases' and 'Databases per page' set to 20, with a page number '1' and navigation arrows.

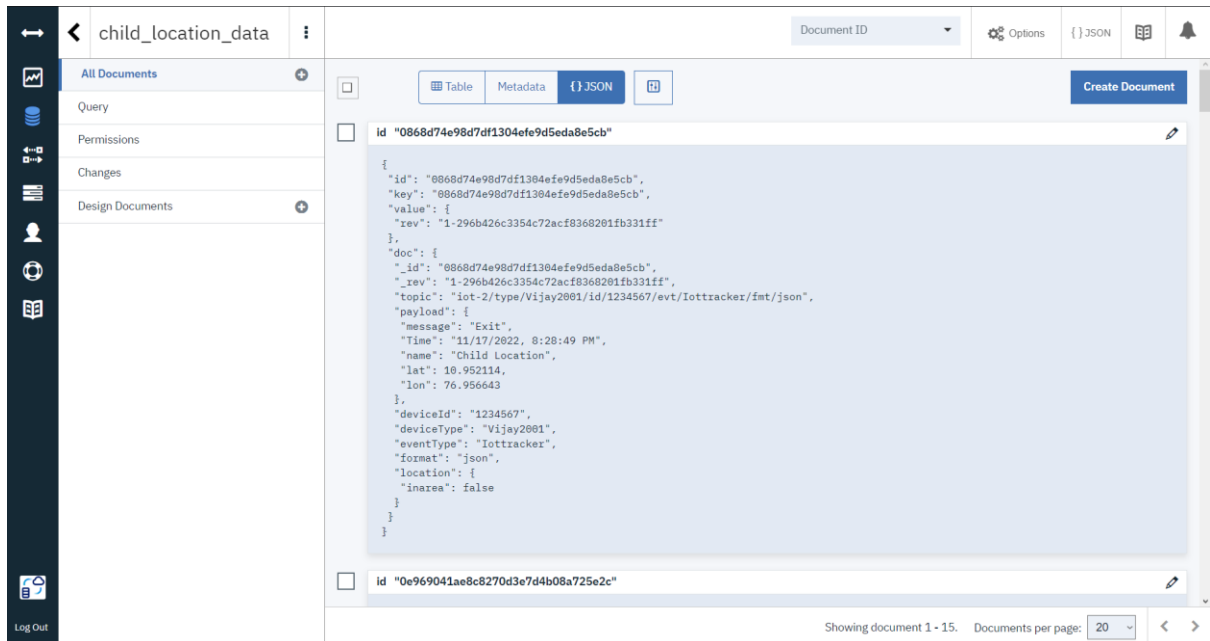
Name	Size	# of Docs	Partitioned	Actions
child_location_data	4.2 KB	15	No	  
noderedfxtyl20221116	33.1 KB	4	No	  

- Store the location coordinates in database



The screenshot shows a web application interface for managing documents. On the left is a dark sidebar with navigation icons and a 'Log Out' button at the bottom. The main header area includes a breadcrumb 'child\_location\_data', a 'Document ID' dropdown, and icons for 'Options', 'JSON', and a notification bell. Below the header, there's a 'Create Document' button and tabs for 'Table', 'Metadata', and 'JSON'. The 'Table' tab is active, displaying a list of documents with columns: \_id, deviceId, deviceType, eventType, and format. The table contains 15 rows of data, all with 'Iottracker' as the event type and 'json' as the format. At the bottom, there's a pagination bar showing 'Showing 5 of 9 columns' and 'Showing document 1 - 15' with a 'Documents per page' dropdown set to 20.

_id	deviceId	deviceType	eventType	format
0868d74e98d7df1304e...	1234567	Vijay2001	Iottracker	json
0e969041ae8c8270d3e...	1234567	Vijay2001	Iottracker	json
157ac8402bfb1ac164e...	1234567	Vijay2001	Iottracker	json
3adc73c6976026de939...	1234567	Vijay2001	Iottracker	json
409f267391f2effd2c49...	1234567	Vijay2001	Iottracker	json
614090b2566fce401c...	1234567	Vijay2001	Iottracker	json
6a2704e478a02b6d7ca...	1234567	Vijay2001	Iottracker	json
6a80894819b94d31a94...	1234567	Vijay2001	Iottracker	json
85eb9cafd6373bd0670...	1234567	Vijay2001	Iottracker	json
9ed31bd378c7e01a9c8...	1234567	Vijay2001	Iottracker	json
bd327c90bb30302890...	1234567	Vijay2001	Iottracker	json
bf8d0ec61d74dc3f53a7...	1234567	Vijay2001	Iottracker	json
c327345e7901f08e4f41...	1234567	Vijay2001	Iottracker	json
d6483b57ff8aaf6c7a17...	1234567	Vijay2001	Iottracker	json

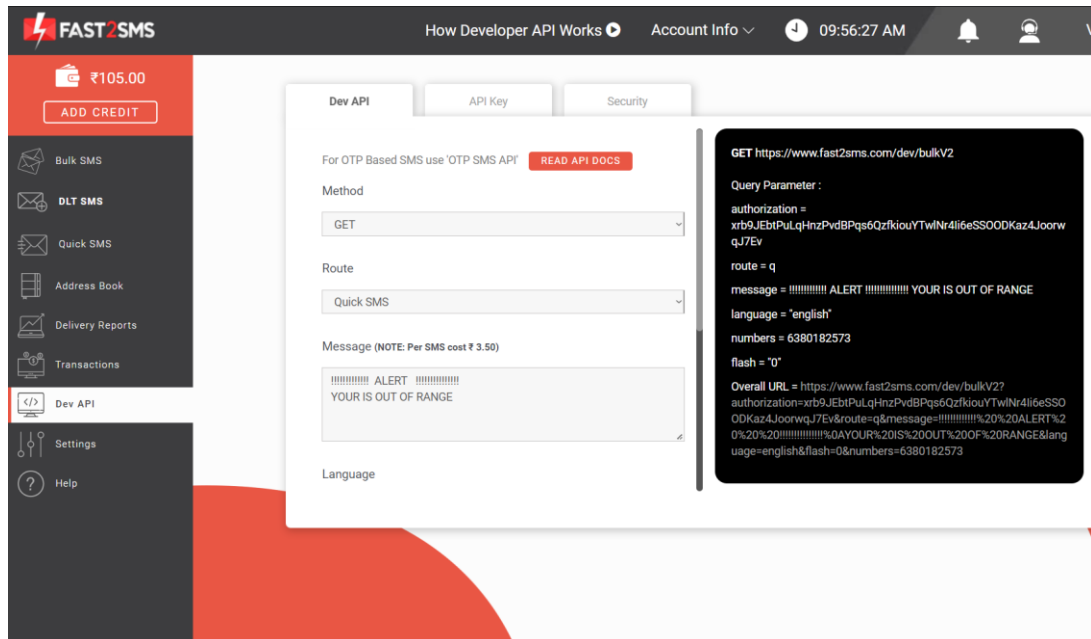


This screenshot shows the same application interface but with the 'JSON' tab selected. It displays the raw JSON data for the first document in the list. The JSON object contains metadata like '\_id', '\_rev', and '\_topic', along with a 'payload' object that includes location coordinates ('lat' and 'lon'), device information, and event details. The interface also shows the document ID and a 'Create Document' button at the top right. The bottom pagination bar indicates 'Showing document 1 - 15' and 'Documents per page: 20'.

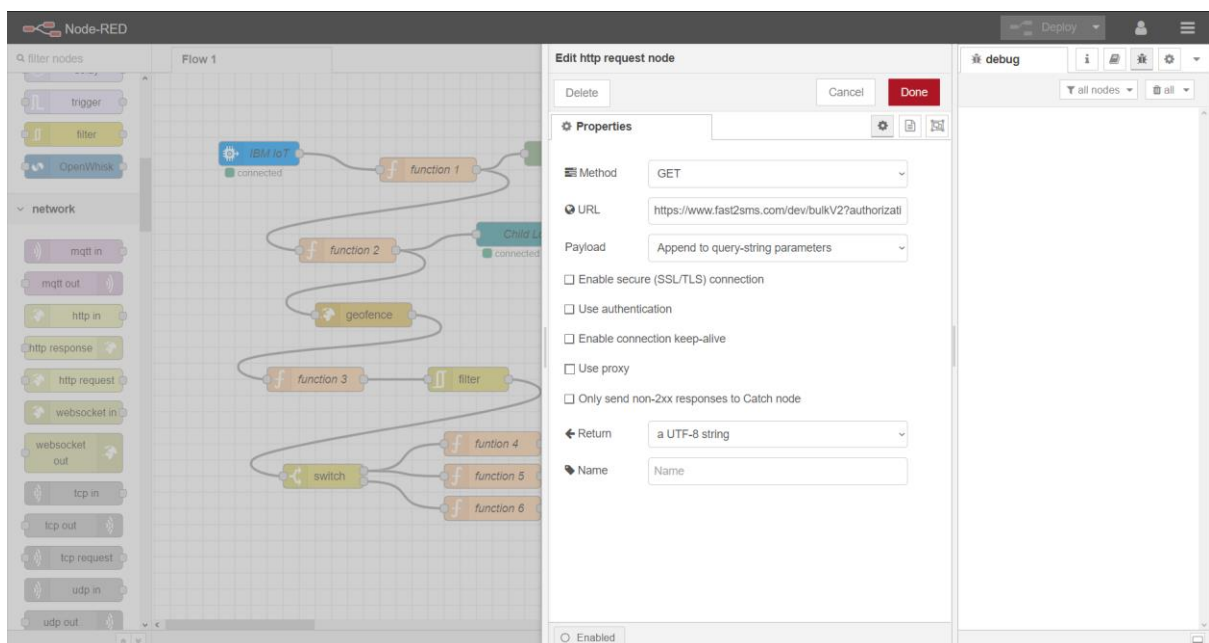
```
{
  "id": "0868d74e98d7df1304efe9d5eda8e5cb",
  "key": "0868d74e98d7df1304efe9d5eda8e5cb",
  "value": {
    "rev": "1-296b426c3354c72acf8368201fb331ff"
  },
  "doc": {
    "_id": "0868d74e98d7df1304efe9d5eda8e5cb",
    "_rev": "1-296b426c3354c72acf8368201fb331ff",
    "topic": "iot-2/type/Vijay2001/id/1234567/evt/Iottracker/fmt/json",
    "payload": {
      "message": "Exit",
      "Time": "11/17/2022, 8:28:49 PM",
      "name": "Child Location",
      "lat": 10.952114,
      "lon": 76.956643
    },
    "deviceId": "1234567",
    "deviceType": "Vijay2001",
    "eventType": "Iottracker",
    "format": "json",
    "location": {
      "inarea": false
    }
  }
}
```

## ALERT SMS:

- Created Dev API ULR to send message through Fast2sms services.



- Paste the url in http request node in node red.



## SMS RECEIVED TO MOBILE PHONE:

