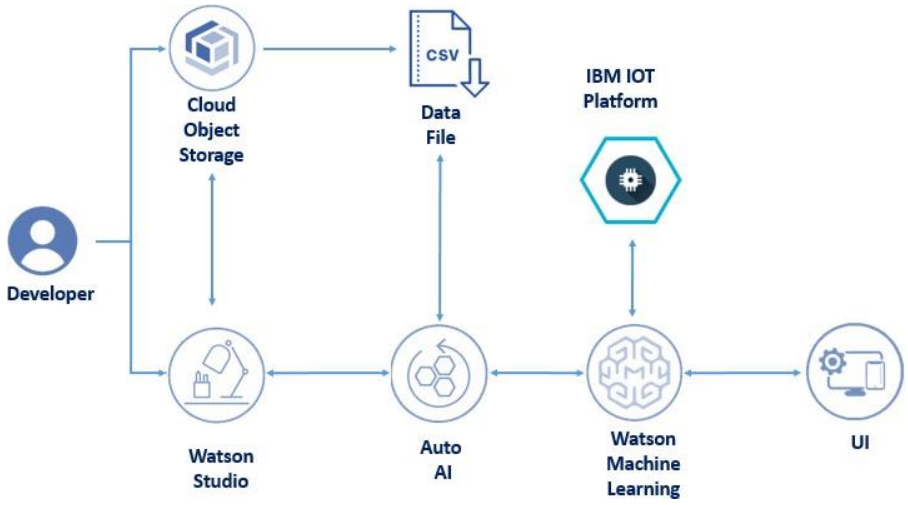
Analysing Sensor Data with machine

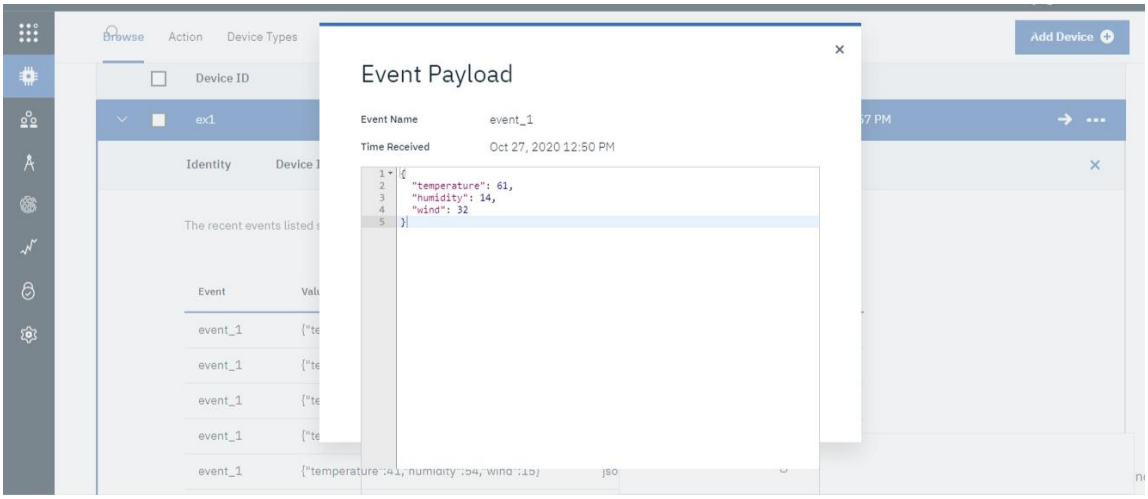
learning The architecture of the project selected is shown below.



The services used in the project are listed below.
IBM Watson Studio
IBM Watson Machine Learning intance
Auto AI with Dataset
IBM Watson IoT Platform
Node Red for building the Application model

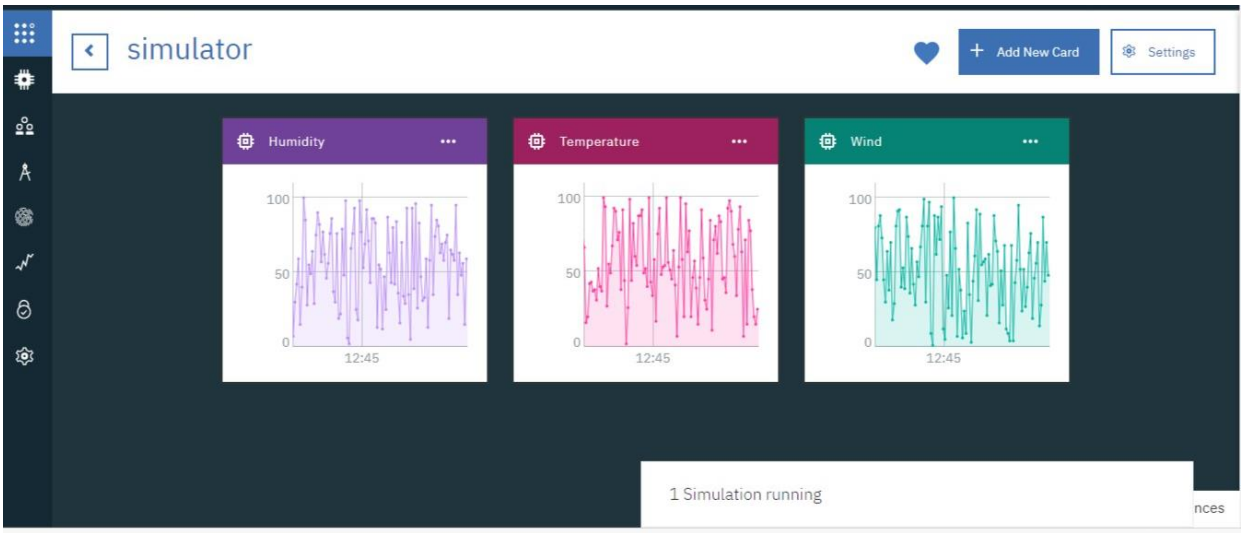
Part 1:
After creating IBM cloud account and log in to the account and the instance of IoT service platform is created as listed by following steps.
Step 1:
Create an instance of IoT service from dashboard by selecting the region as Dallas.
Step 2:
Click on the IoT instance from services list in dashboard. It directs to IoT platform window, click launch and create the device of IoT filling the data as per the instructions given in session and run the device with simulator and copy the data to dashboards and cards. The device credentials are created and are stored in notepad.
The link of the simulated output from boards and cards is given below.
https://wy3fg4.internetofthings.ibmcloud.com/dashboard/boards/62696e84-b970-43bd-97c5-d452ce9073dd

Connecting the device to internal sensor and repeat visualizing with boards and cards as per the required sensor used.
Now connect the required internal sensor by turning on the simualtion button of the device and schedule the 0 simlation running window to required number of records per minute in event 1 simulation 1 window. And add the dashboards and cards for this event.
The results of internal simulator run and visualised in dashboards through boards and cards as labelled are shown in below listed images with link of all specified below.



[Iot event payload with device simulator tuned ON]

[Boards and Cards for IoT sensor with its simulator ON]



/

Part 2:

Building AI/ML Model:

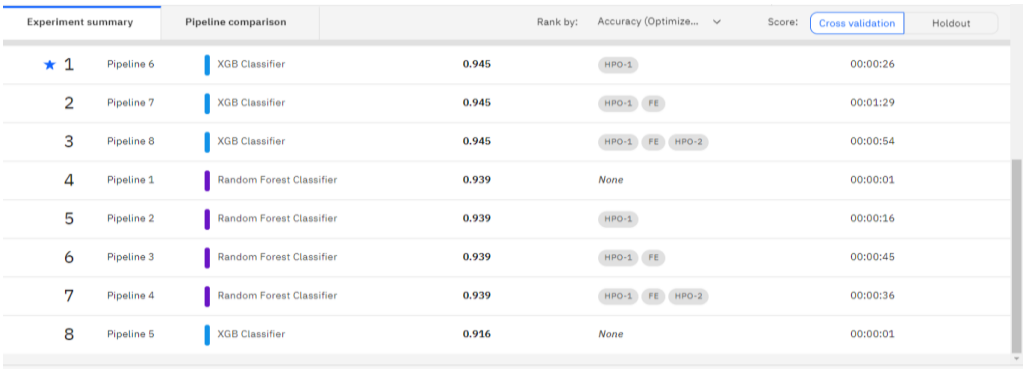
Step 1:

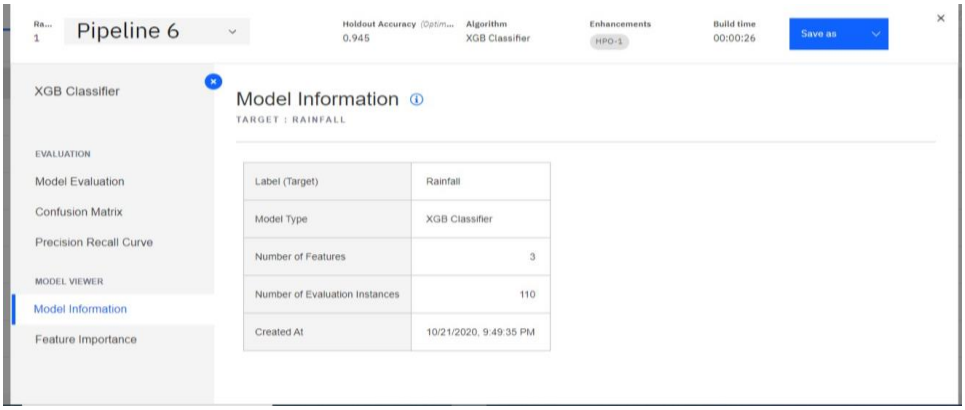
Select Watson Studio and create new project and ML instance is created adding storage object along with it.

To this ML instance Auto AI experiment service is added and dataset as given by the mentor is uploaded and the experiment is run with the parameter selected based on the project output parameter. Now the model developed with high rated star is stored which is to be deployed.

Step 2:

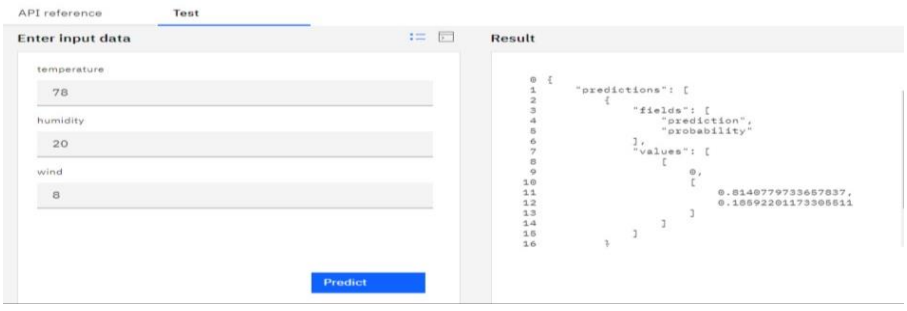
After successful completion of AI experiment, the model stored in step 1 above is deployed creating a space and storing it in the same storage object space as that of machine language instance. Once the model deployment is completed, the API reference of endpoints of the model are noted to notepad and also the python code is stored. The AutoAi experiment after completion with pipeline comparison and model information details are listed are shown in the pictures uploaded. From these we can get information about the algorithm used in training the ML model and the pipiline number which is rated high is listed at top in the log information. this is the one to be stored in order to deploy it into the space which will consume lower storage due to its efficient prediction and analysis of the dataset.





Step 3:

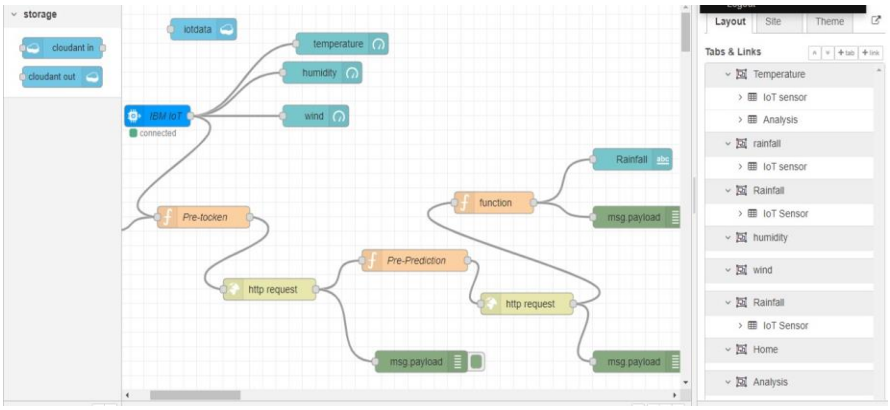
Now the model deployed is tested by going to the labelled deployment space from dashboard and test tab is clicked and the input parameters are entered in input column of the window and after pressing the Predict button, Result column on right side displays the python code of the output.

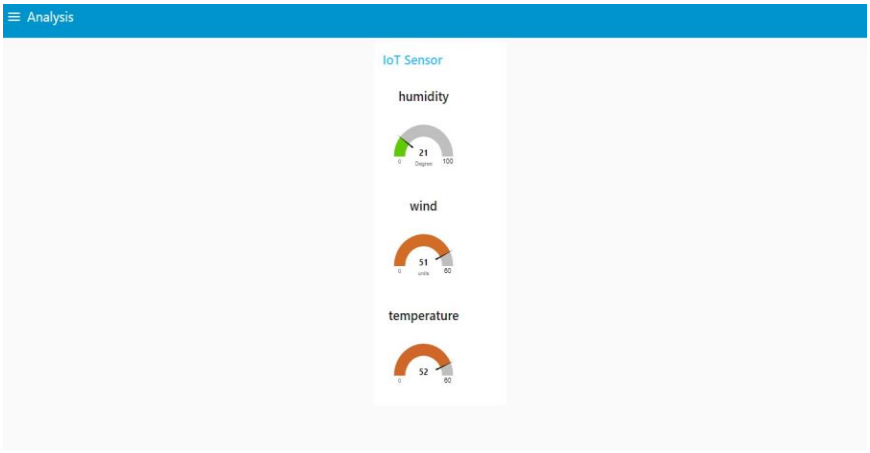


/

Part 3:

In this stage the node Red application installed is opened in new tab with all other required tabs of IBM being active to access them in between while building the application model of the project. After installing all the required dashboard and ibm nodes from the palette of the node red application, the required flow is imported from the file given. Then the IBM cloud API key generated is assigned in IBM IoT input node and by following the instructions given in the training session the nodes are modified as per the project architecture flow. The output obtained is shown through images enclosed.





/