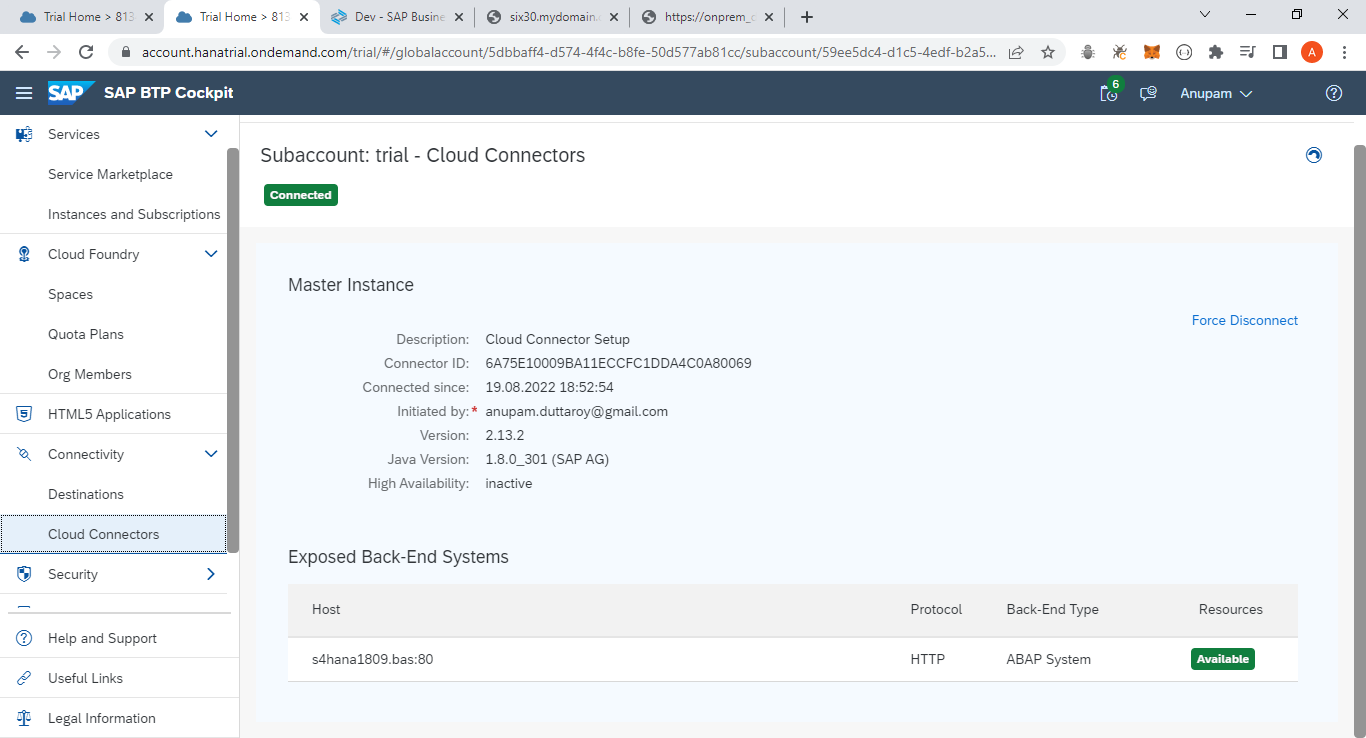
Cloud Connector



Destination

Graphical user interface, text, application

Description automatically generated

App Deployed in CF using the destination name to fetch data

Graphical user interface, text, application, email

Description automatically generated

Index.js in Nodejs application at BAS:

const express = require('express');

const cfAxios = require('sap-cf-axios').default;

const app = express();

const port = process.env.VCAP\_APP\_PORT || 5000;

app.get('/onprem', async (req, res) => {

    axios\_cf = cfAxios('S4H\_1809');

    const response = await axios\_cf({

        method : 'GET',

        url: '/sap/opu/odata/sap/ZTEST\_ODATA1\_SRV/StudentSet',

        params: {

            $format : 'json'

        },

        header : {

            accept: 'application/json'

        }

    });

    console.log(response);

    res.send(response.data.d.results);

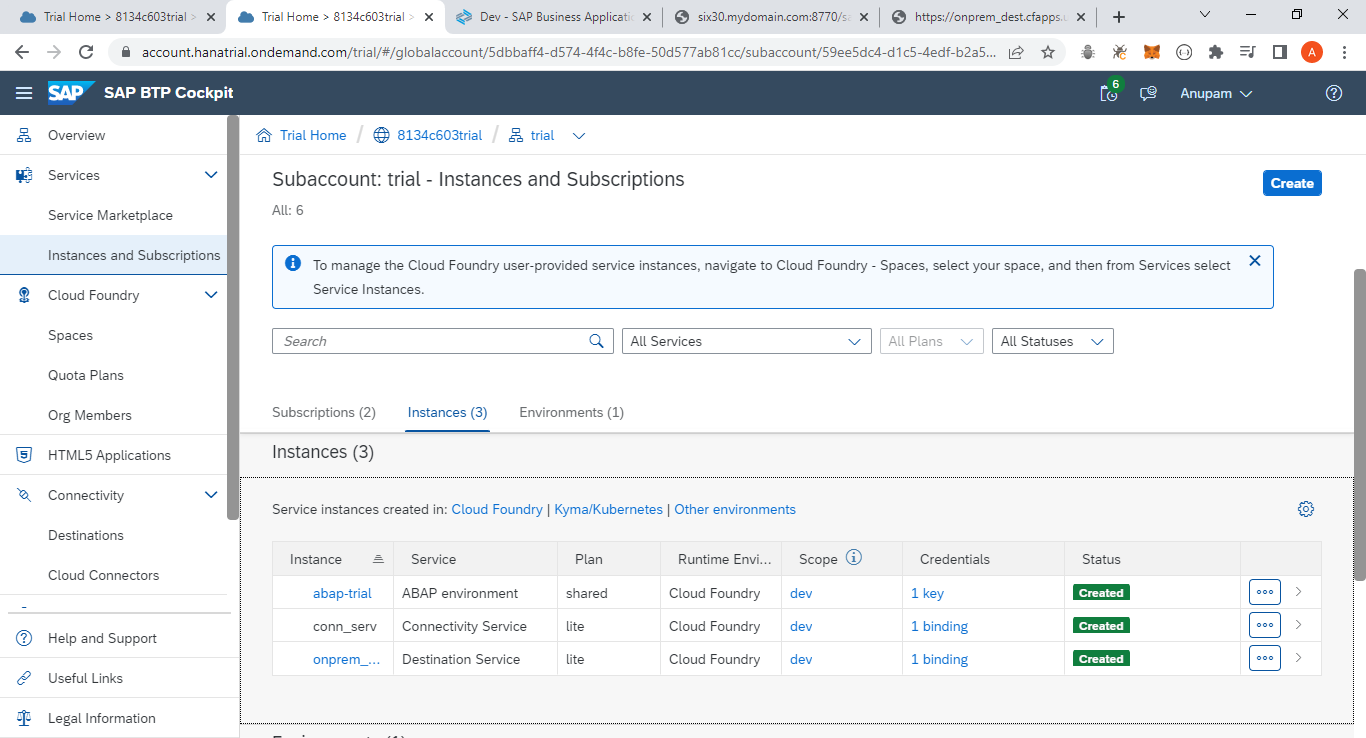
})

app.listen(port, () => {

    console.log('App is running on port ', port);

})

Subscribed services from service marketplace: Destination Service and Connectivity Service.



Both the services have binding with App service created in CF:

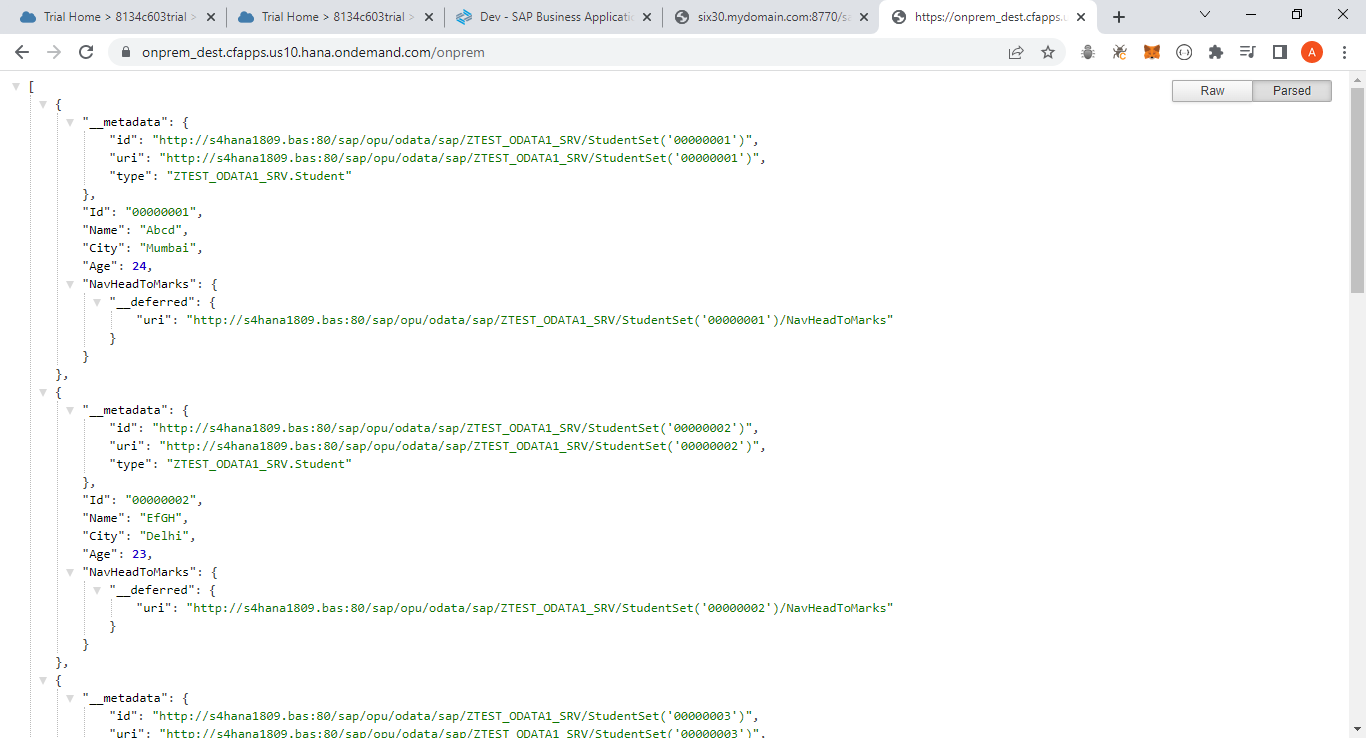
Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Data fetched from on-prem:



To work without destination, direct call to the endpoint to be made with authorization details:

const express = require('express');

const axios = require('axios');

const app = express();

const port = process.env.VCAP\_APP\_PORT || 5000;

app.get('/onprem', async (req, res) => {

    var config = {

      method: 'get',

      url: 'http://103.44.1.51:8770/sap/opu/odata/sap/ZTEST\_ODATA1\_SRV/StudentSet',

      headers: {

        'Cookie': 'SAP\_SESSIONID\_S20\_800=5EPMbJ0H9mJM-uqLuae10jo2K7EhxhHtorEYZtr1Xno%3d; sap-usercontext=sap-client=800'

      },

       auth: {

           username: '', //Use SAP username

           password: '' // Use SAP password

       }

    };

    axios(config)

    .then(function (response) {

      console.log(JSON.stringify(response.data));

      res.send(response.data.d.results);

    })

    .catch(function (error) {

      console.log(error);

    });

})

app.listen(port, () => {

    console.log('App is running on port ', port);

})