

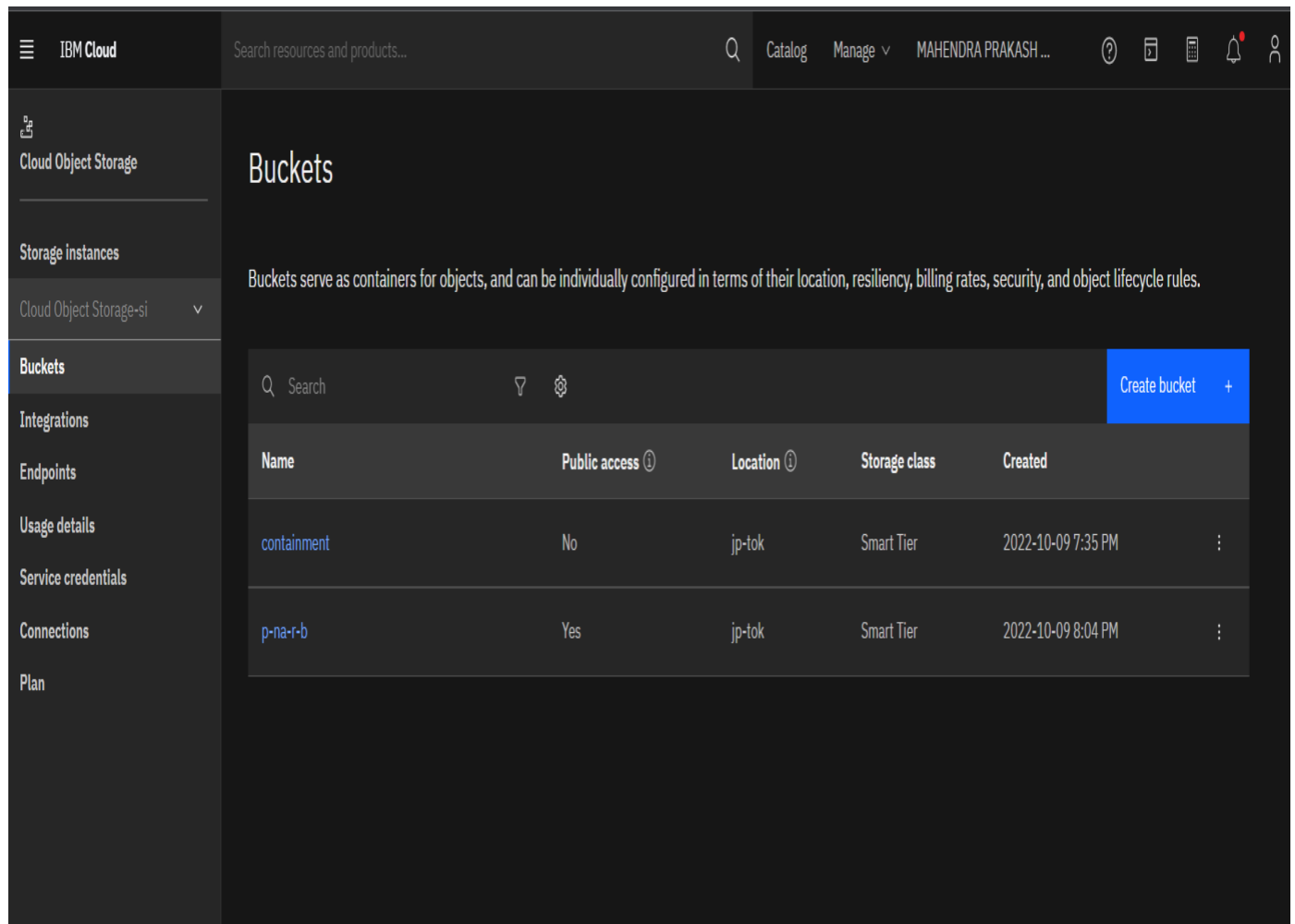
ASSIGNMENT 3

TEAM ID : PNT2022TMID50559

PROJECT NAME : CONTAINMENT ZONE ALERTING APPLICATION

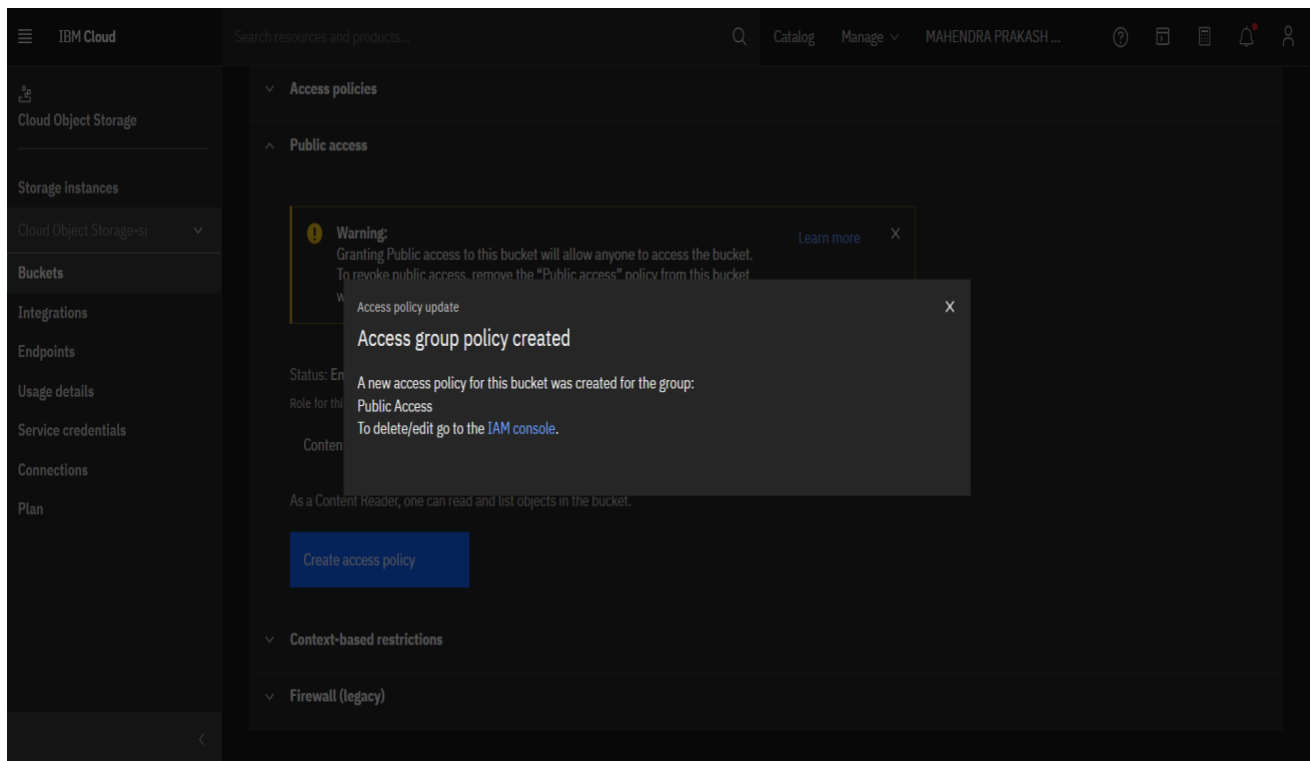
STUDENT NAME : A.JANNATH FIRTHOUSE

Q1. Creating a Bucket in IBM Cloud Storage



The screenshot displays the IBM Cloud Storage interface. On the left is a navigation sidebar with options: Cloud Object Storage, Storage instances, Cloud Object Storage-si, Buckets (selected), Integrations, Endpoints, Usage details, Service credentials, Connections, and Plan. The main content area is titled 'Buckets' and includes a description: 'Buckets serve as containers for objects, and can be individually configured in terms of their location, resiliency, billing rates, security, and object lifecycle rules.' Below this is a table of buckets with columns: Name, Public access, Location, Storage class, and Created. A 'Create bucket' button is visible in the top right of the table area.

Name	Public access ⓘ	Location ⓘ	Storage class	Created
containment	No	jp-tok	Smart Tier	2022-10-09 7:35 PM
p-na-r-b	Yes	jp-tok	Smart Tier	2022-10-09 8:04 PM



Q2. Uploading 5 images to IBM Object storage and made it public

<!DOCTYPE html>

<html>

<head>

<title>IMAGES</title>

<link rel="stylesheet" href="https://mani07.s3.jp-tok.cloud-object-storage.appdomain.cloud/stylenew.css">

</head>

<body>

<h2>GALLERY</h2>

```
<h3>EXPLORE THE BEAUTY ...</h3>

<div id="pic">

</div>

</body>

</html>
```

Q3:Upload a css page to the object storage and use the same page in your HTML code.

```
body{

    background-color:rgb(115, 82, 190);

}

#pic{

    margin-top: 10%;

    margin: 5%;

    line-height:0;

    column-count:5;
```

```

        column-gap:5px;
        background-image: rgb(83, 6, 134);
    }
    #pic img{
        width: 100%!important;
        height: auto !important;
        margin-bottom:auto;
    }
    h2{
        text-align: center;
        font-family: sans-serif;
        font-size: 50px;
        margin-top: 3%;
        font-weight: bolder;
        color: rgb(233, 241, 234);
    }
    h3{
        text-align:center;
        font-family: sans-serif;
        font-size: 50px;
        font-weight: bolder;
        color:rgb(249, 247, 252);
    }

```

Q4:Design a chatbot using IBM Watson assistant for hospital.

```

<!Doctype
html>

```

```
<html>
  <head>
    <title>Chat Boat</title>
    <style>
      body{
        background-color: rgb(160, 233, 158);
      }
      img{
        width: 200px;
        height: 100px;
        margin-left: 10px;
      }
      h2{
        text-align:center;
        color: rgba(71, 9, 218, 0.89);
        font-size:4pc;
      }
      h3{
        text-align:center;
        color:rgb(238, 6, 141);
        font-size:2.5pc;
      }
    </style>
  </head>
  <body>
    <h2>HOSPITALS</h2>
    <h2></h2>
```

```

    <p><h3>WE ARE PROVIDING THE DETAILS FOR YOUR
    QUERIES</h3>

    <h3> Click the chatbot </h3>

  </p>

  <script>

    window.watsonAssistantChatOptions = {

      integrationID: "4f8894e3-1817-41e3-bf8b-a5d4b9b2197a", // The ID
      of this integration.

      region: "au-syd", // The region your integration is hosted in.

      serviceInstanceID: "ba115cc6-e2d6-4d6b-beea-74977285d64c", //
      The ID of your service instance.

      onLoad: function(instance) { instance.render(); }

    };

    setTimeout(function(){

      const t=document.createElement('script');

      t.src="https://web-
      chat.global.assistant.watson.appdomain.cloud/versions/" +
      (window.watsonAssistantChatOptions.clientVersion || 'latest') +
      "/WatsonAssistantChatEntry.js";

      document.head.appendChild(t);

    });

  </script>

  </body>

</html>

```

Q5.Create Watson assistant service with 10 steps and use 3 conditions in it. Load that script in HTML page.

```

<html>

<body>

  <script>

```

```
window.watsonAssistantChatOptions = {  
  integrationID: "ec603862-aa33-4336-a203-961870431228", // The ID  
  of this integration.  
  region: "au-syd", // The region your integration is hosted in.  
  serviceInstanceID: "44f5354b-4c14-4c36-bdfa-6fddf1e73eaa", // The ID  
  of your service instance.  
  onLoad: function(instance) { instance.render(); }  
};  
setTimeout(function(){  
  const t=document.createElement('script');  
  t.src="https://web-  
chat.global.assistant.watson.appdomain.cloud/versions/" +  
(window.watsonAssistantChatOptions.clientVersion || 'latest') +  
"/WatsonAssistantChatEntry.js";  
  document.head.appendChild(t);  
});  
</script>  
</body>  
</html>
```