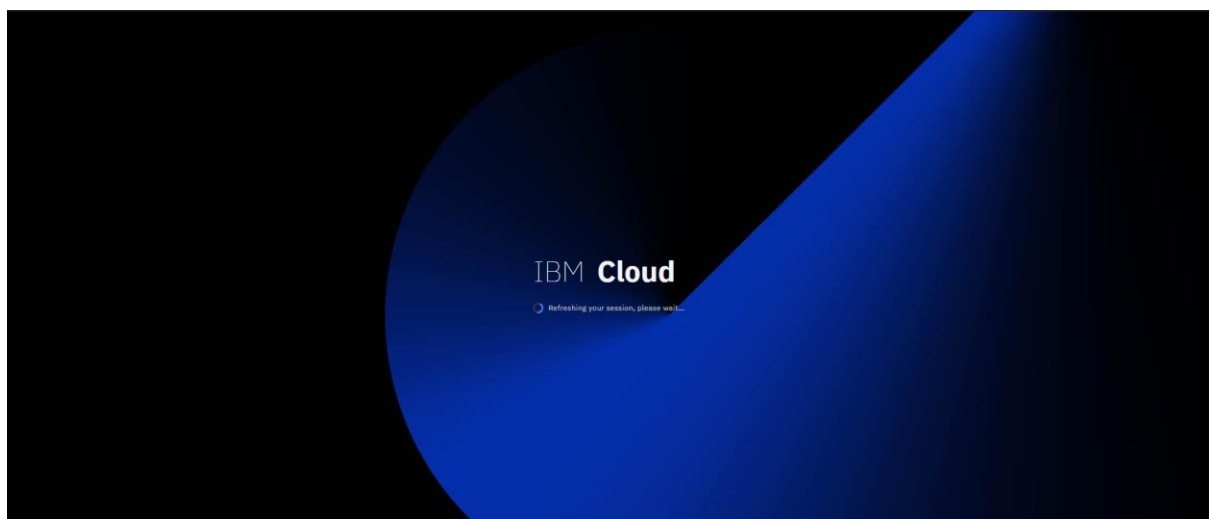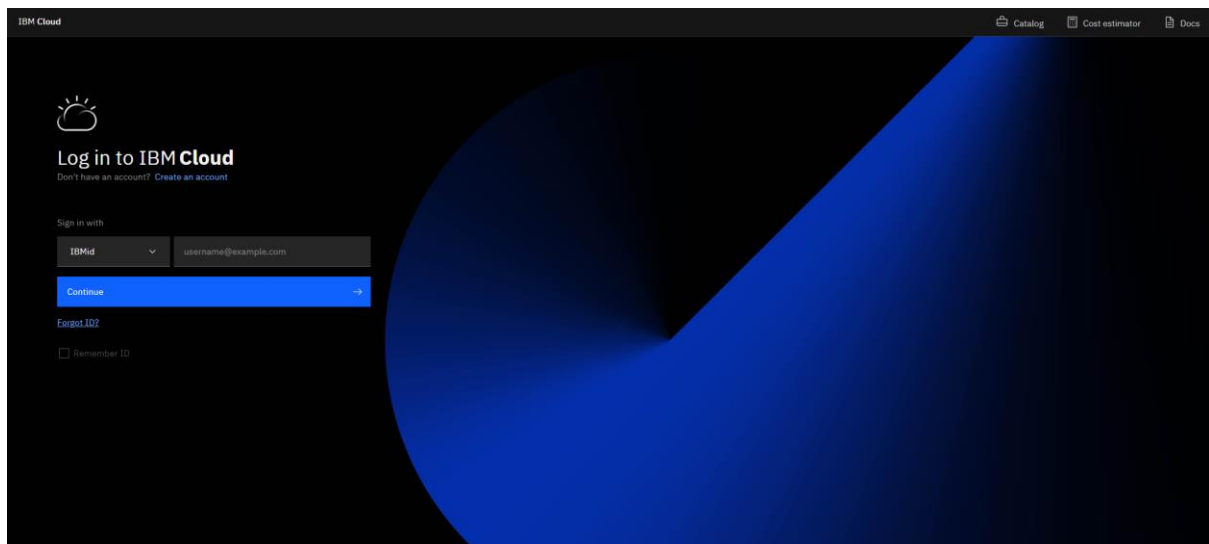**CAD – PHASE 3 ASSIGNMENT**

**DEVELOPMENT PHASE 1**

**Design a simple web interface where users can upload images and view the AI-generated captions.**

Here are the step-by-step instructions to create a simple web interface for users to upload images and view AI-generated captions.

**Step 1: Create an IBM Cloud Account**

**Step 2: Log into IBM Cloud**

### Step 3: Set Up the Visual Recognition Service

- Once you are logged in, go to the IBM Cloud Dashboard.
- Click on "Create Resource" to create a new service.
- In the search bar, type "Visual Recognition" and select the "Visual Recognition" service from the catalog.
- Follow the prompts to create the service. You may need to choose a region and give your service a name.

### Step 4: Obtain API Keys

### Step 5: Design a Simple Web Interface

```html
<!DOCTYPE html>
<html>
<head>
  <title>Image Recognition</title>
</head>
<body>
  <h1>Image Recognition</h1>
  <input type="file" id="imageInput" accept="image/*" />
  <button onclick="uploadImage()">Upload Image</button>
  <div id="imageContainer"></div>
  <div id="captionContainer"></div>

  <script>
    const apiKey = 'YOUR_API_KEY';
    const apiUrl = 'YOUR_API_URL';

    function uploadImage() {
      const fileInput = document.getElementById('imageInput');
```

```javascript
        const imageContainer = document.getElementById('imageContainer');
        const captionContainer = document.getElementById('captionContainer');


        if (fileInput.files.length > 0) {
            const imageFile = fileInput.files[0];


            const formData = new FormData();
            formData.append('images_file', imageFile);


            fetch(apiUrl + '/v3/classify?version=2022-01-01', {
                method: 'POST',
                headers: {
                    'Authorization': 'Basic ' + btoa('apikey:' + apiKey)
                },
                body: formData
            })
            .then(response => response.json())
            .then(data => {
                // Display the uploaded image
                const imageUrl = URL.createObjectURL(imageFile);
                const imgElement = document.createElement('img');
                imgElement.src = imageUrl;
                imageContainer.innerHTML = '';
                imageContainer.appendChild(imgElement);


                // Display the AI-generated captions
```

```
            captionContainer.innerHTML = '<strong>AI-Generated
Captions:</strong><br>';

                data.images[0].classifiers[0].classes.forEach(cls => {
                    captionContainer.innerHTML += cls.class + ': ' + cls.score +
'<br>';
                });
            })
        .catch(error => {
            console.error('Error:', error);
        });
        } else {
            alert('Please select an image to upload.');
        }
    }
    </script>
</body>
</html>
```

**Step 6: Integrate the Visual Recognition API**

**Watson Studio-pq**  ● Active   Add tags ✎

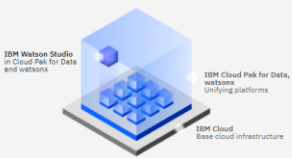Details   Actions... ⌄

Manage
Plan

**Watson Studio in**
**Cloud Pak for Data and watsonx**

Build and deploy machine learning models on either platform.
Work with foundation models on watsonx as a Service.

Launch in ⌄

IBM Watson Studio
in Cloud Pak for Data
and watsonx

IBM Cloud Pak for Data,
watsonx
Unifying platforms

IBM Cloud
Base cloud infrastructure

IBM Watson Studio is part of IBM Cloud Pak for Data and watsonx, and
serves as the AI capability of the data fabric architecture.

**Helpful links**

**Documentation**

Learn about tools, features, and how to perform a wide variety of Data
and AI tasks.

Cloud Pak for Data  →
watsonx  →

**Learning path**

Start a step-by-step tutorial to get up and running quickly.

Cloud Pak for Data  →
watsonx  →

**Videos**

Watch videos to learn about Watson Studio.

Cloud Pak for Data  →
watsonx  →

**How to use Watson Studio**

Build, deploy, and trust AI models

---

Take a tutorial

Step through implementing a Data
fabric use case in a sample
project.
→

Quick start

Build customer profiles
with IBM Match 360 with Watson

Catalog and govern data
with Watson Knowledge Catalog

Build and manage ML models
with Watson Studio

Query data anywhere
with Watson Query

What's new

Connect to more data sources in DataStage
Oct 27, 2023

Use a Satellite Connector to connect to an on-prem database
Oct 27, 2023

New in gallery

Mini_Daily_Temperatures

This dataset describes the minimum daily temperatures over
10 years (1981-1990) in the city Melbourne, Australia. The

**Build and manage ML models**
with Watson Studio

Watson Studio is a service that you use to build, deploy, and
manage AI models and to optimize decisions.
Work within a project to build models.
Customize how you work by choosing from notebooks, graphical
canvases, and no-code tools.

**Get started**

Sample project
Open a sample project with pre-built Watson Studio
assets.

New project
Create a project and then add your own data to get
started.

Cancel        Next

Google Chrome

---

## New project

**Define details**

Name
web interface - AI GENERATED CAPTIONS

Description (optional)
What's the purpose of this project?

**Controls**

☑ Restrict who can be a collaborator ⓘ
☐ Mark as sensitive ⓘ

**Storage**

Project includes integration with Cloud Object Storage for storing project assets.

Cloud Object Storage-rg

Cancel        Create

## General

Associated Service : watson-vision-combined-px

Overview | **Test** | Implementation

**Filter**

Threshold      0.0

0 ●———— 1

**Classes**
- [ ] animal
- [ ] ash grey color
- [ ] ball
- [ ] blue
- [ ] bottle green color
- [ ] building
- [ ] car
- [ ] carnivore
- [ ] clothing
- [ ] clothing store

✕ Clear results

**animal2.jpg**



| giant panda | 0.99 |
| carnivore | 0.99 |
| mammal | 0.99 |
| animal | 0.99 |
| greenishness color | 0.69 |
| indigo color | 0.35 |

**ball.jpg**



| tennis ball | 1.00 |
| ball | 1.00 |
| bottle green color | 0.88 |
| lemon yellow color | 0.80 |

**billgates.jpg**



| ash grey color | 0.94 |
| person | 0.70 |
| decision maker | 0.56 |
| official | 0.52 |
| Secretary of the Interior | 0.51 |
| Treasury | 0.50 |

---

## General

Associated Service : watson-vision-combined-px

Overview | **Test** | Implementation

**Filter**

Threshold      0.0

0 ●———— 1

**Classes**
- [ ] adult person
- [ ] animal
- [x] ash grey color
- [ ] ball
- [ ] bar
- [ ] beige color
- [ ] Bengal tiger
- [x] big cat
- [ ] blue
- [ ] bottle green color

**Shoes1.jpg**



| ash grey color | 0.99 |
| foot pedal | 0.84 |
| lever | 0.84 |
| bar | 0.84 |
| sports equipment | 0.80 |
| clothing | 0.78 |
| footwear | 0.70 |

**tiger.jpg**



| animal | 0.98 |
| mammal | 0.97 |
| carnivore | 0.97 |
| feline | 0.97 |
| big cat | 0.97 |
| tiger | 0.78 |
| light brown color | 0.75 |