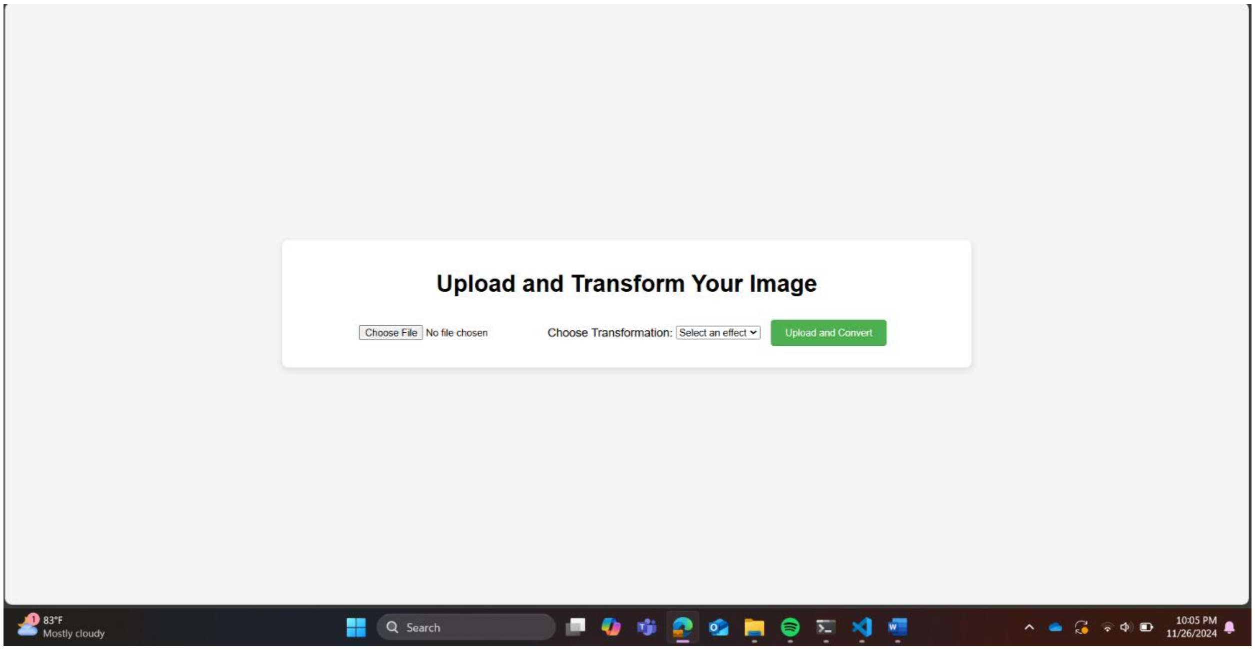
Nama: Terrence Reinhardt

NIM: 2602182581

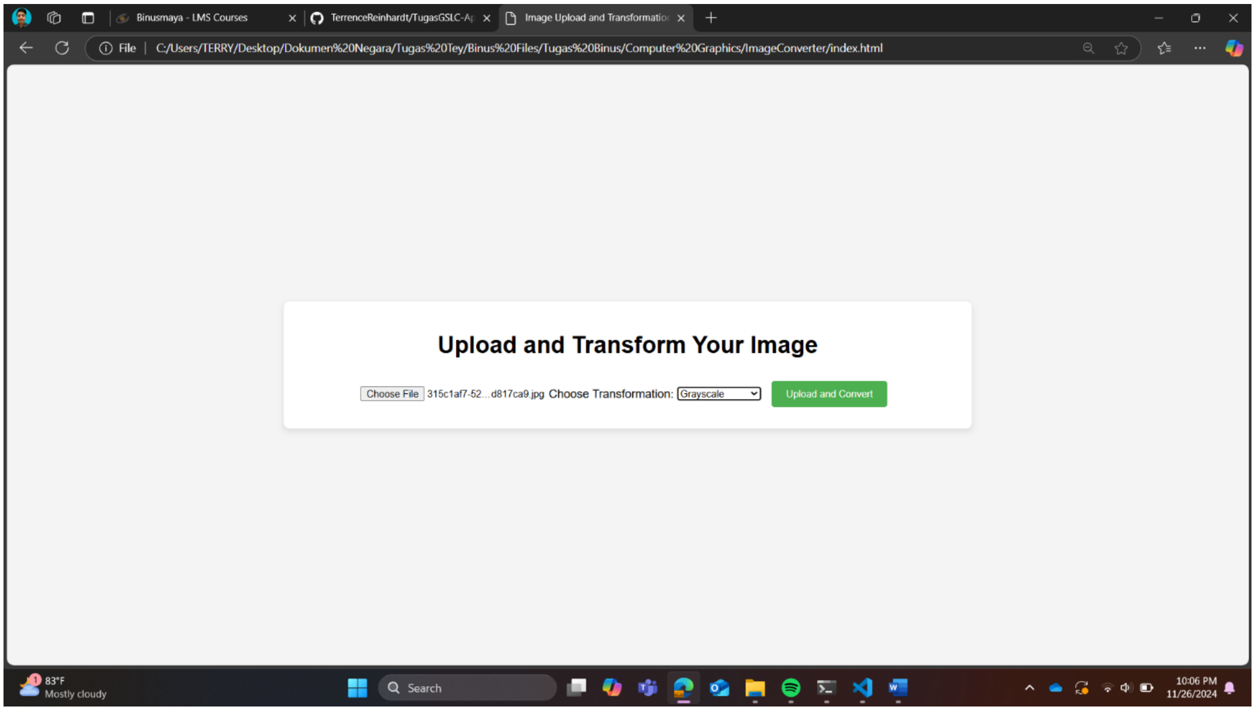
Tugas GSLC Computer Graphics

Output:

1. Tampilan Awal



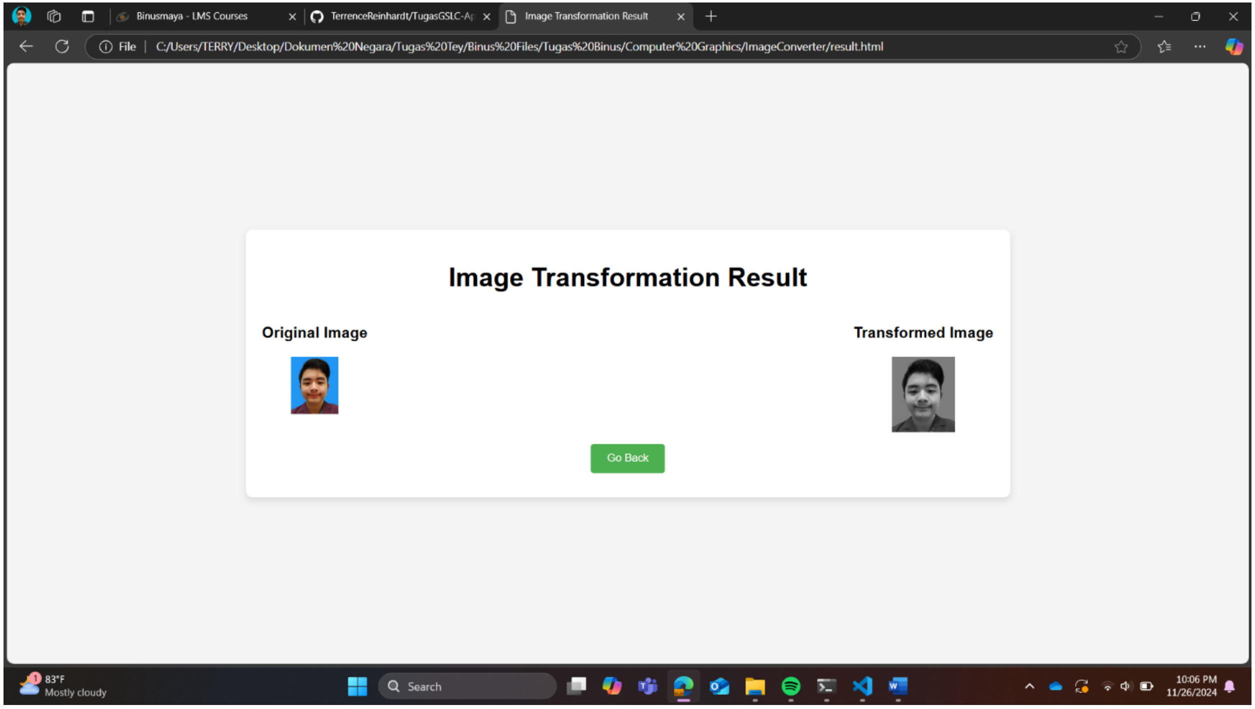
2. Mau upload image ke grayscale beserta hasilnya



File | C:/Users/TERRY/Desktop/Dokumen%20Negara/Tugas%20Tey/Binus%20Files/Tugas%20Binus/Computer%20Graphics/ImageConverter/index.html

Upload and Transform Your Image

Choose File 315c1af7-52...d817ca9.jpg Choose Transformation: Grayscale



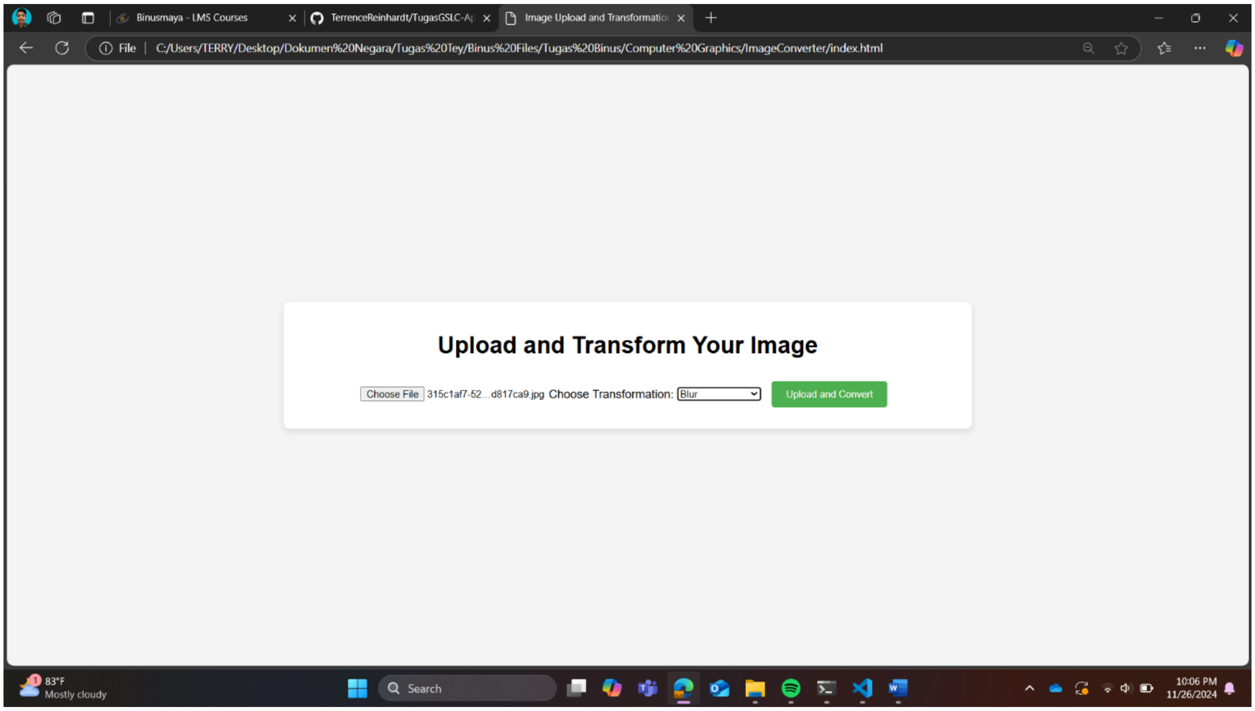
File | C:/Users/TERRY/Desktop/Dokumen%20Negara/Tugas%20Tey/Binus%20Files/Tugas%20Binus/Computer%20Graphics/ImageConverter/result.html

Image Transformation Result

Original Image

Transformed Image

3. Sama kayak no 2 cuman imagenya diblur beserta hasilnya


Binusmaya - LMS Courses
TerrenceReinhardt/TugasGSLC-A
Image Upload and Transformatio
File
C:/Users/TERRY/Desktop/Dokumen%20Negara/Tugas%20Tey/Binus%20Files/Tugas%20Binus/Computer%20Graphics/ImageConverter/index.html

Upload and Transform Your Image

Choose File 315c1af7-52...d817ca9.jpg Choose Transformation: Blur Upload and Convert

83°F
Mostly cloudy
Search
10:06 PM
11/26/2024

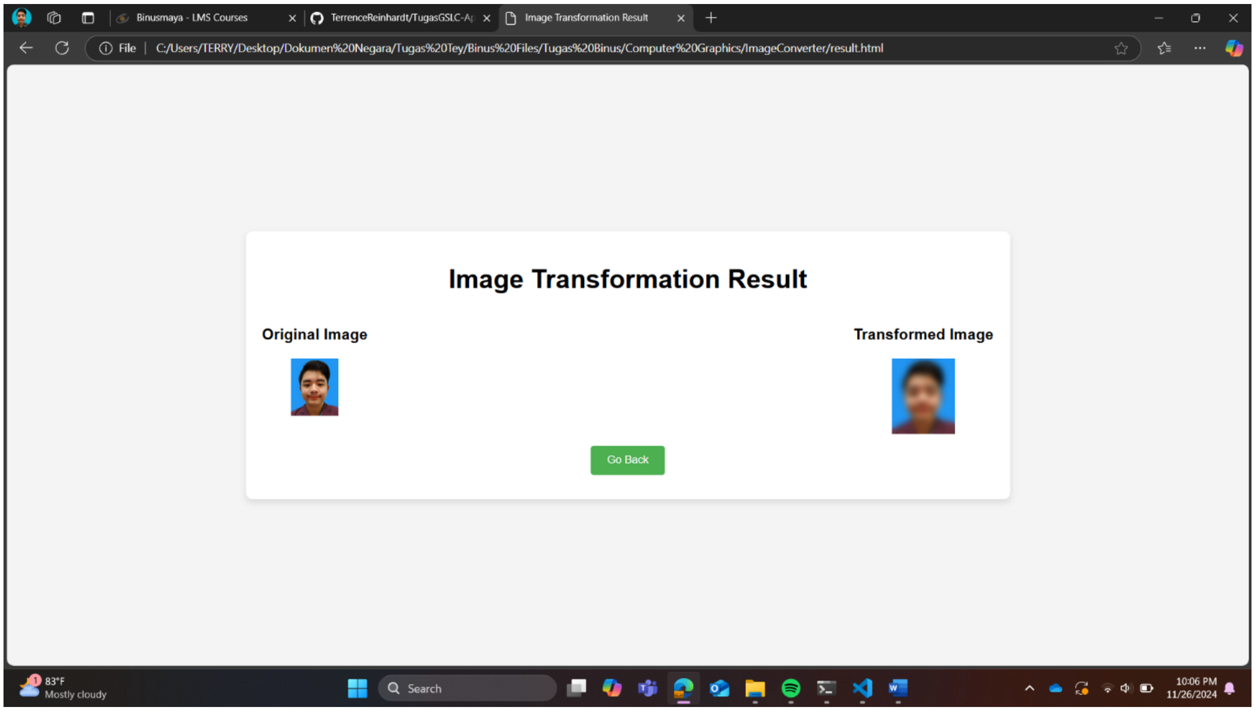

Binusmaya - LMS Courses
TerrenceReinhardt/TugasGSLC-A
Image Transformation Result
File
C:/Users/TERRY/Desktop/Dokumen%20Negara/Tugas%20Tey/Binus%20Files/Tugas%20Binus/Computer%20Graphics/ImageConverter/result.html

Image Transformation Result

Original Image

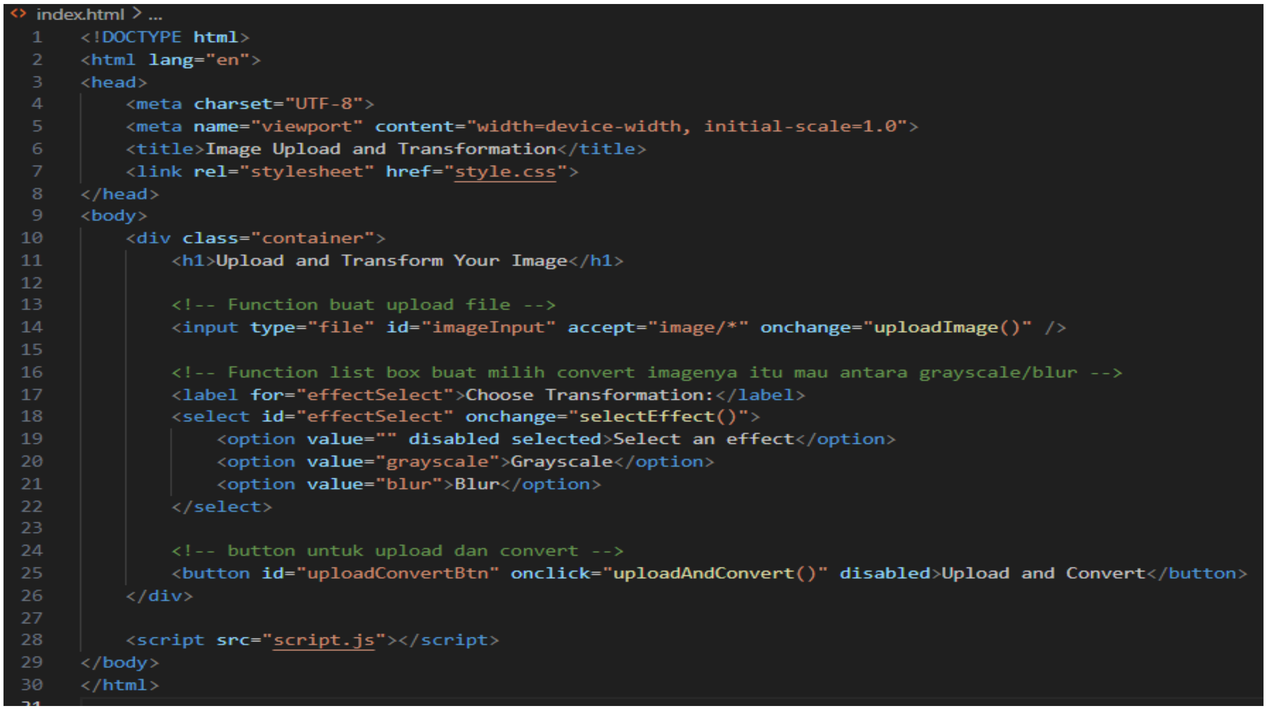
Transformed Image

Go Back

83°F
Mostly cloudy
Search
10:06 PM
11/26/2024

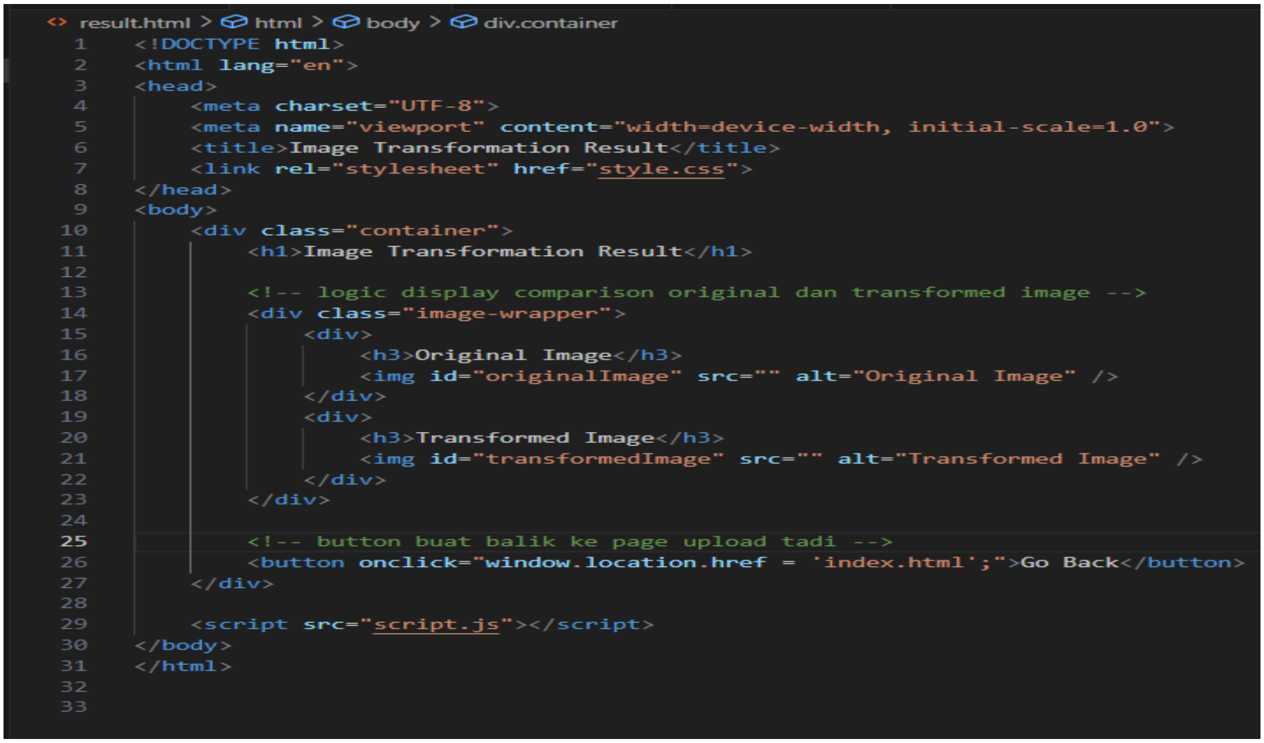
Code (Penjelasannya saya kasih comment di tiap function code dalam gambar2 berikut):

1. index.html



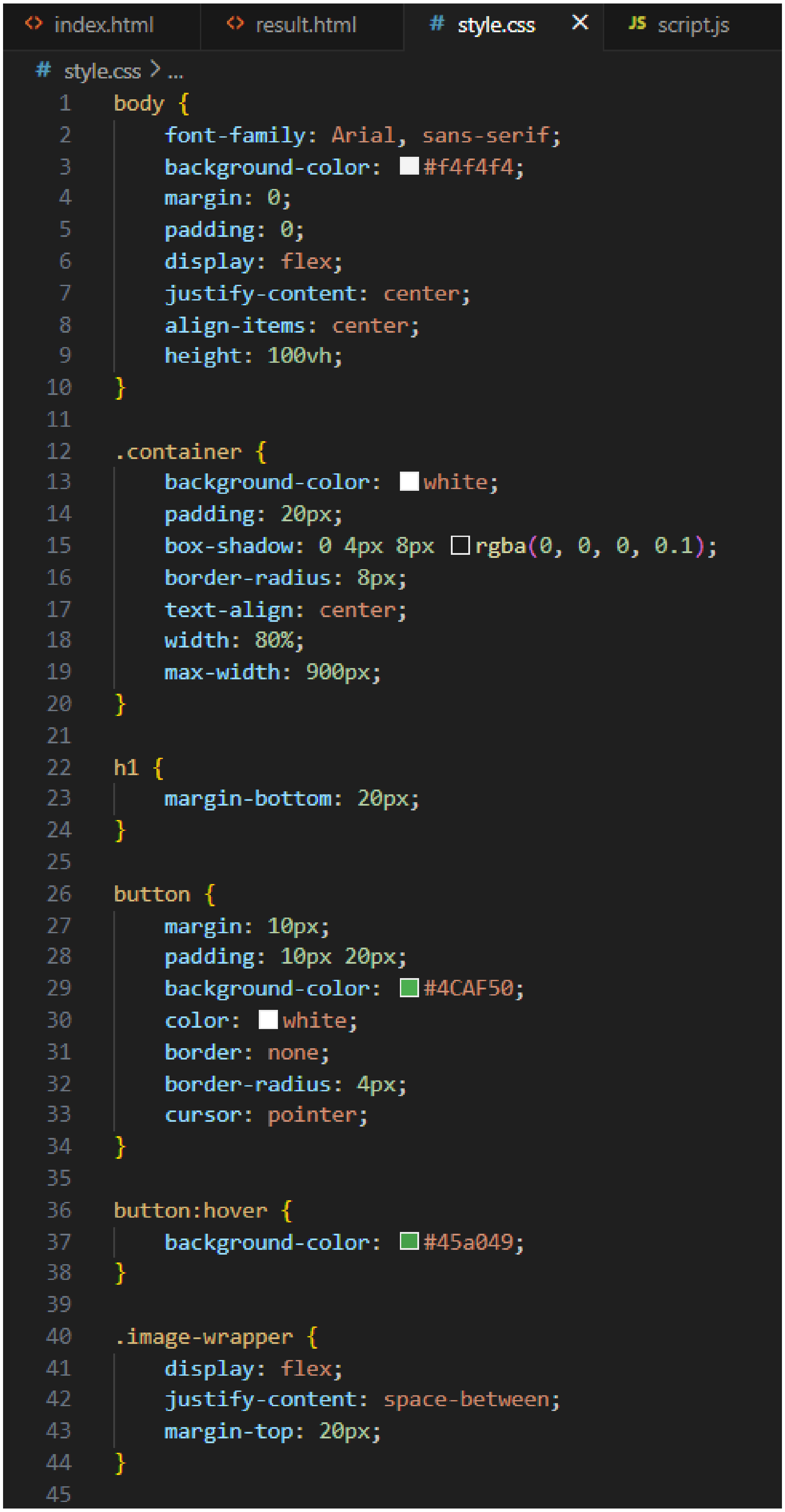
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Image Upload and Transformation</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="container">
        <h1>Upload and Transform Your Image</h1>

        <!-- Function buat upload file -->
        <input type="file" id="imageInput" accept="image/*" onchange="uploadImage()" />

        <!-- Function list box buat milih convert imagenya itu mau antara grayscale/blur -->
        <label for="effectSelect">Choose Transformation:</label>
        <select id="effectSelect" onchange="selectEffect()">
            <option value="" disabled selected>Select an effect</option>
            <option value="grayscale">Grayscale</option>
            <option value="blur">Blur</option>
        </select>

        <!-- button untuk upload dan convert -->
        <button id="uploadConvertBtn" onclick="uploadAndConvert()" disabled>Upload and Convert</button>
    </div>

    <script src="script.js"></script>
</body>
</html>
```

2. result.html

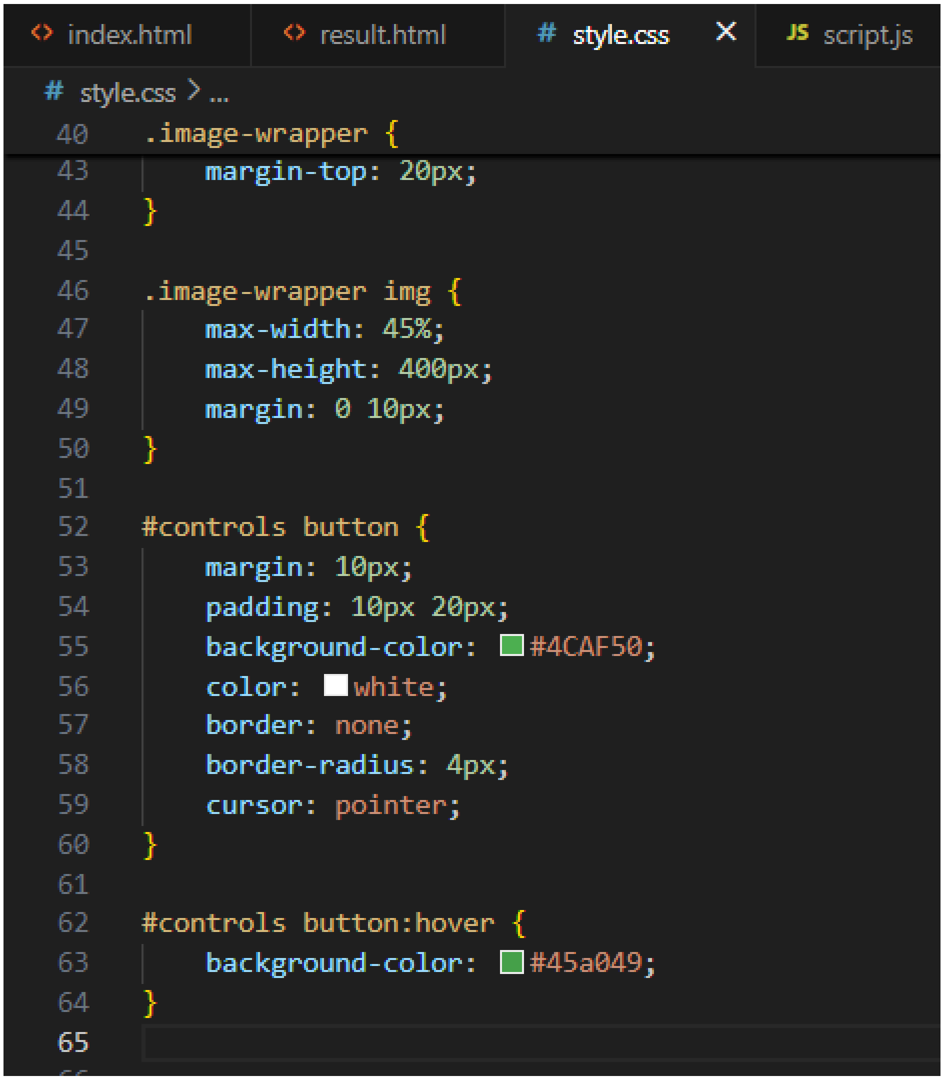


```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Image Transformation Result</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="container">
        <h1>Image Transformation Result</h1>

        <!-- logic display comparison original dan transformed image -->
        <div class="image-wrapper">
            <div>
                <h3>Original Image</h3>
                <img id="originalImage" src="" alt="Original Image" />
            </div>
            <div>
                <h3>Transformed Image</h3>
                <img id="transformedImage" src="" alt="Transformed Image" />
            </div>
        </div>

        <!-- button buat balik ke page upload tadi -->
        <button onclick="window.location.href = 'index.html';">Go Back</button>
    </div>

    <script src="script.js"></script>
</body>
</html>
```

3. style.css

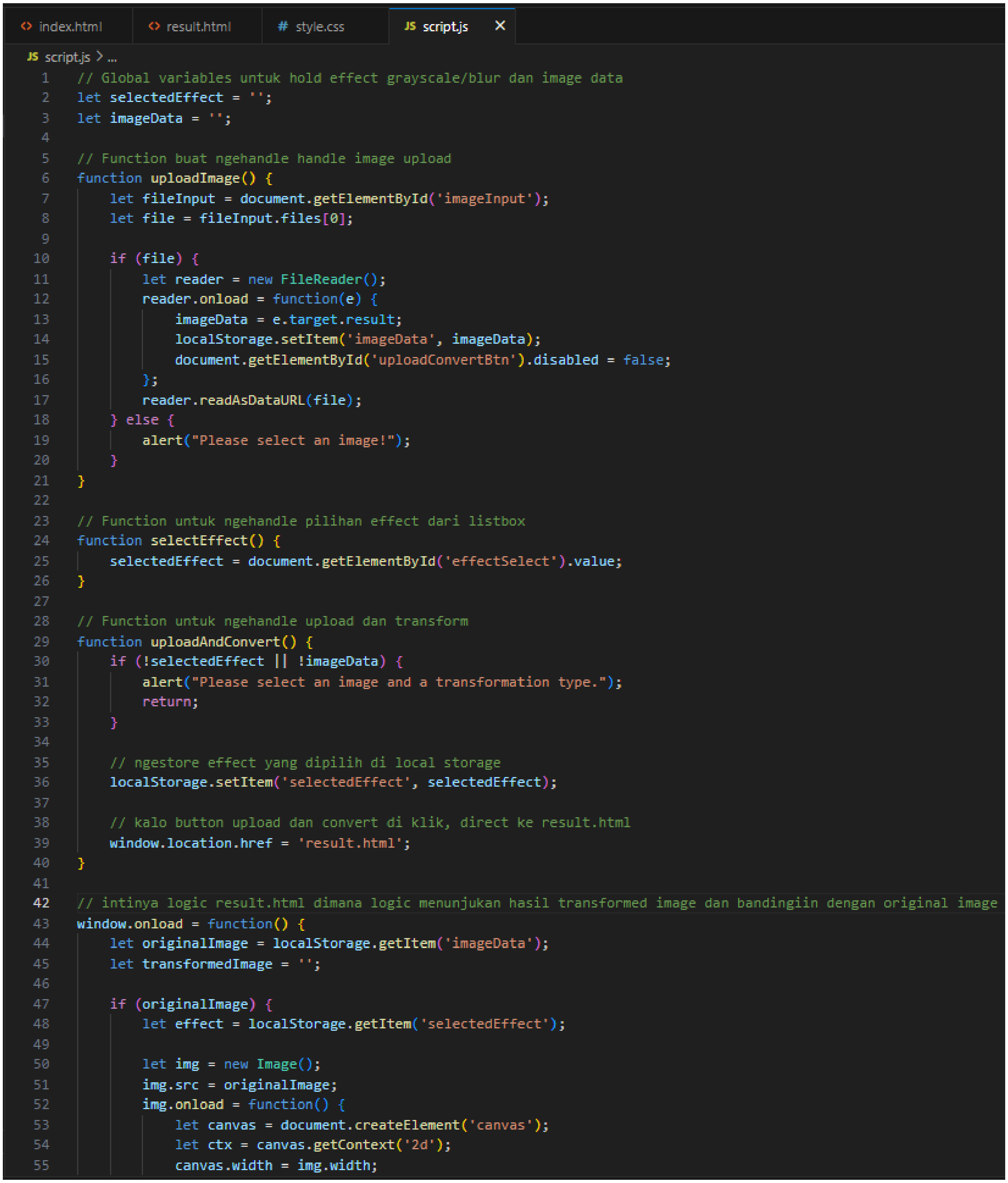


```css
body {
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
    margin: 0;
    padding: 0;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
}

.container {
    background-color: white;
    padding: 20px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
    border-radius: 8px;
    text-align: center;
    width: 80%;
    max-width: 900px;
}

h1 {
    margin-bottom: 20px;
}

button {
    margin: 10px;
    padding: 10px 20px;
    background-color: #4CAF50;
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

button:hover {
    background-color: #45a049;
}

.image-wrapper {
    display: flex;
    justify-content: space-between;
    margin-top: 20px;
}

```

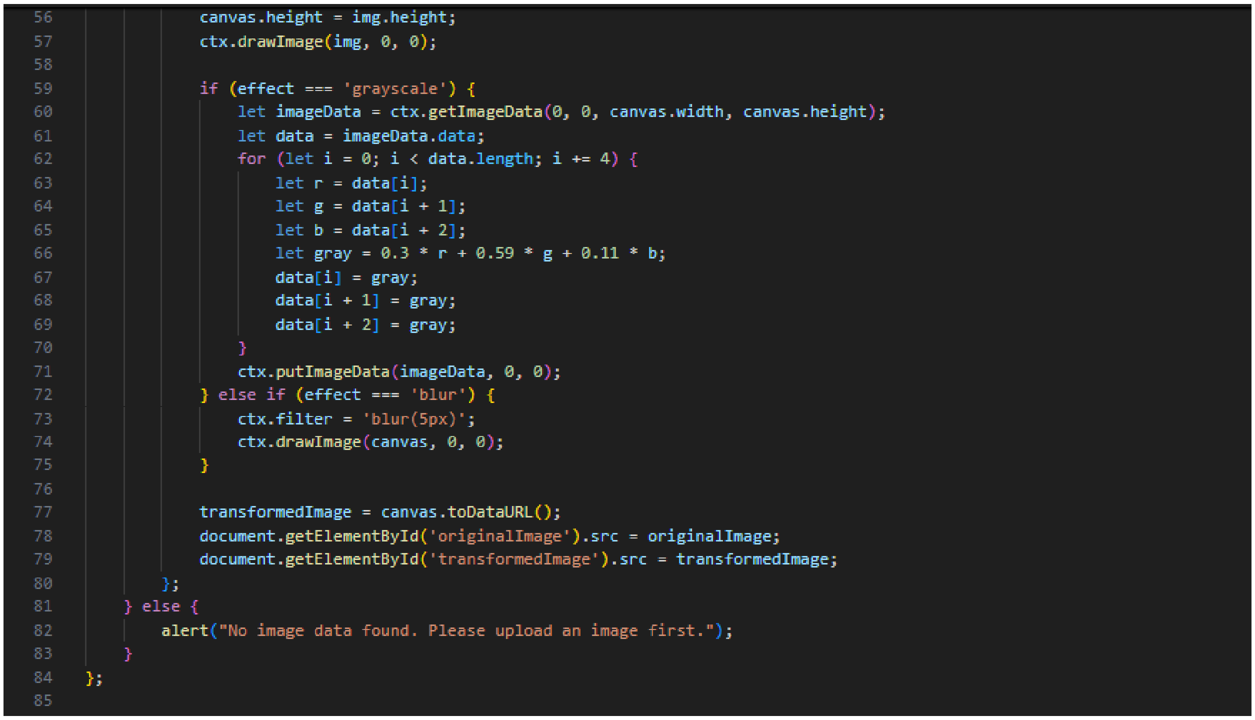


```css
.image-wrapper {
    margin-top: 20px;
}

.image-wrapper img {
    max-width: 45%;
    max-height: 400px;
    margin: 0 10px;
}

#controls button {
    margin: 10px;
    padding: 10px 20px;
    background-color: #4CAF50;
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

#controls button:hover {
    background-color: #45a049;
}

```

4. script.js

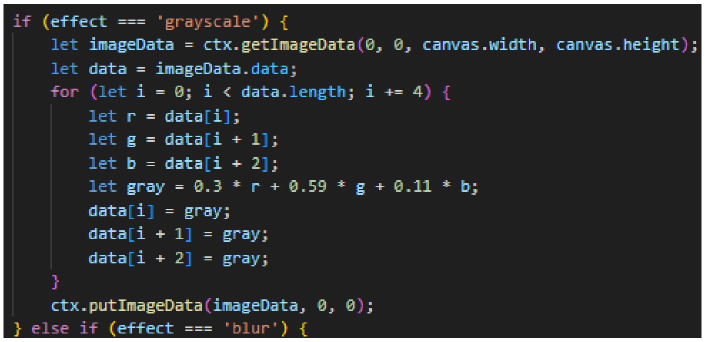


```javascript
// Global variables untuk hold effect grayscale/blur dan image data
let selectedEffect = '';
let imageData = '';

// Function buat ngehandle handle image upload
function uploadImage() {
    let fileInput = document.getElementById('imageInput');
    let file = fileInput.files[0];

    if (file) {
        let reader = new FileReader();
        reader.onload = function(e) {
            imageData = e.target.result;
            localStorage.setItem('imageData', imageData);
            document.getElementById('uploadConvertBtn').disabled = false;
        };
        reader.readAsDataURL(file);
    } else {
        alert("Please select an image!");
    }
}

// Function untuk ngehandle pilihan effect dari listbox
function selectEffect() {
    selectedEffect = document.getElementById('effectSelect').value;
}

// Function untuk ngehandle upload dan transform
function uploadAndConvert() {
    if (!selectedEffect || !imageData) {
        alert("Please select an image and a transformation type.");
        return;
    }

    // ngestore effect yang dipilih di local storage
    localStorage.setItem('selectedEffect', selectedEffect);

    // kalo button upload dan convert di klik, direct ke result.html
    window.location.href = 'result.html';
}

// intinya logic result.html dimana logic menunjukan hasil transformed image dan bandingiin dengan original image
window.onload = function() {
    let originalImage = localStorage.getItem('imageData');
    let transformedImage = '';

    if (originalImage) {
        let effect = localStorage.getItem('selectedEffect');

        let img = new Image();
        img.src = originalImage;
        img.onload = function() {
            let canvas = document.createElement('canvas');
            let ctx = canvas.getContext('2d');
            canvas.width = img.width;
```



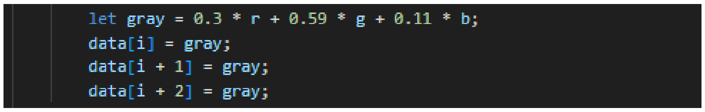
```
            canvas.height = img.height;
            ctx.drawImage(img, 0, 0);

            if (effect === 'grayscale') {
                let imageData = ctx.getImageData(0, 0, canvas.width, canvas.height);
                let data = imageData.data;
                for (let i = 0; i < data.length; i += 4) {
                    let r = data[i];
                    let g = data[i + 1];
                    let b = data[i + 2];
                    let gray = 0.3 * r + 0.59 * g + 0.11 * b;
                    data[i] = gray;
                    data[i + 1] = gray;
                    data[i + 2] = gray;
                }
                ctx.putImageData(imageData, 0, 0);
            } else if (effect === 'blur') {
                ctx.filter = 'blur(5px)';
                ctx.drawImage(canvas, 0, 0);
            }

            transformedImage = canvas.toDataURL();
            document.getElementById('originalImage').src = originalImage;
            document.getElementById('transformedImage').src = transformedImage;
        };
    } else {
        alert("No image data found. Please upload an image first.");
    }
};

```

Contoh perhitungan:



```
if (effect === 'grayscale') {
    let imageData = ctx.getImageData(0, 0, canvas.width, canvas.height);
    let data = imageData.data;
    for (let i = 0; i < data.length; i += 4) {
        let r = data[i];
        let g = data[i + 1];
        let b = data[i + 2];
        let gray = 0.3 * r + 0.59 * g + 0.11 * b;
        data[i] = gray;
        data[i + 1] = gray;
        data[i + 2] = gray;
    }
    ctx.putImageData(imageData, 0, 0);
} else if (effect === 'blur') {
```

Jika kita lihat contoh function grayscale tersebut, dijelaskan cara logic tersebut mengkonversi image yang telah diupload ke grayscale. Jika diliat, declarement "let gray" merupakan logic perubahan warna2 rgb menjadi grayscale.



```
        let gray = 0.3 * r + 0.59 * g + 0.11 * b;
        data[i] = gray;
        data[i + 1] = gray;
        data[i + 2] = gray;
```

0.3, 0.59 dan 0.11 merupakan adjustment kontribusi dari respective red, green dan blue. Kalo ditulis jadi rumus:
Gray= 0.3×R + 0.59×G + 0.11×B