// components/ImageUploader.js

Import { useState, useEffect } from ‘react’;

Export default function ImageUploader({ onImageChange }) {

Const [image, setImage] = useState(null);

Const handleImageUpload = (event) => {

Const file = event.target.files[0];

Const imageUrl = URL.createObjectURL(file);

setImage(imageUrl);

onImageChange(imageUrl);

};

Return (

<div>

<input type=”file” accept=”image/\*” onChange={handleImageUpload} />

<br />

{image && (

<img

Id=”uploaded-image”

Src={image}

Alt=”Preview”

Style={{ maxWidth: ‘300px’, marginTop: ‘20px’ }}

/>

)}

</div>

);

}

// components/PredictionResult.js

Export default function PredictionResult({ predictions }) {

Return (

<div style={{ marginTop: ‘20px’ }}>

{predictions.map((pred, i) => (

<div key={i}>

{pred.className} — {(pred.probability \* 100).toFixed(2)}%

</div>

))}

</div>

);

}

// pages/index.js

Import { useState, useEffect } from ‘react’;

Import \* as mobilenet from ‘@tensorflow-models/mobilenet’;

Import ‘@tensorflow/tfjs’;

Import ImageUploader from ‘../components/ImageUploader’;

Import PredictionResult from ‘../components/PredictionResult’;

Export default function Home() {

Const [model, setModel] = useState(null);

Const [image, setImage] = useState(null);

Const [predictions, setPredictions] = useState([]);

useEffect(() => {

mobilenet.load().then(setModel);

}, []);

Const recognizeImage = async () => {

If (!model || !image) return;

Const imgElement = document.getElementById(‘uploaded-image’);

Const results = await model.classify(imgElement);

setPredictions(results);

};

Return (

<div style={{ padding: ‘20px’, textAlign: ‘center’ }}>

<h1>AI Image Recognition</h1>

<ImageUploader onImageChange={setImage} />

<button onClick={recognizeImage} style={{ marginTop: ‘10px’ }}>Recognize</button>

<PredictionResult predictions={predictions} />

</div>

);

}

// pages/about.js

Export default function About() {

Return (

<div style={{ padding: ‘20px’, textAlign: ‘center’ }}>

<h1>About</h1>

<p>This AI-powered web app uses TensorFlow.js and React to recognize objects in images.</p>

</div>

);

}

// pages/contact.js

Export default function Contact() {

Return (

<div style={{ padding: ‘20px’, textAlign: ‘center’ }}>

<h1>Contact</h1>

<p>Contact us at: [support@aiapp.com</p](mailto:support@aiapp.com%3c/p)>

</div>

);

}

// pages/services.js

Export default function Services() {

Return (

<div style={{ padding: ‘20px’, textAlign: ‘center’ }}>

<h1>Services</h1>

<p>We offer AI-powered image analysis and computer vision services.</p>

</div>

);

}

// pages/\_app.js

Import ‘../styles/globals.css’;

Export default function App({ Component, pageProps }) {

Return <Component {…pageProps} />;

}

// styles/globals.css

Body {

Margin: 0;

Font-family: Arial, sans-serif;

Background-color: #f5f5f5;

}

H1 {

Color: #333;

}

Button {

Background-color: #4CAF50;

Border: none;

Color: white;

Padding: 10px 20px;

Text-align: center;

Font-size: 16px;

Border-radius: 8px;

Cursor: pointer;

}

// package.json

{

“name”: “ai-image-recognition”,

“version”: “1.0.0”,

“scripts”: {

“dev”: “next dev”,

“build”: “next build”,

“start”: “next start”

},

“dependencies”: {

“next”: “latest”,

“react”: “latest”,

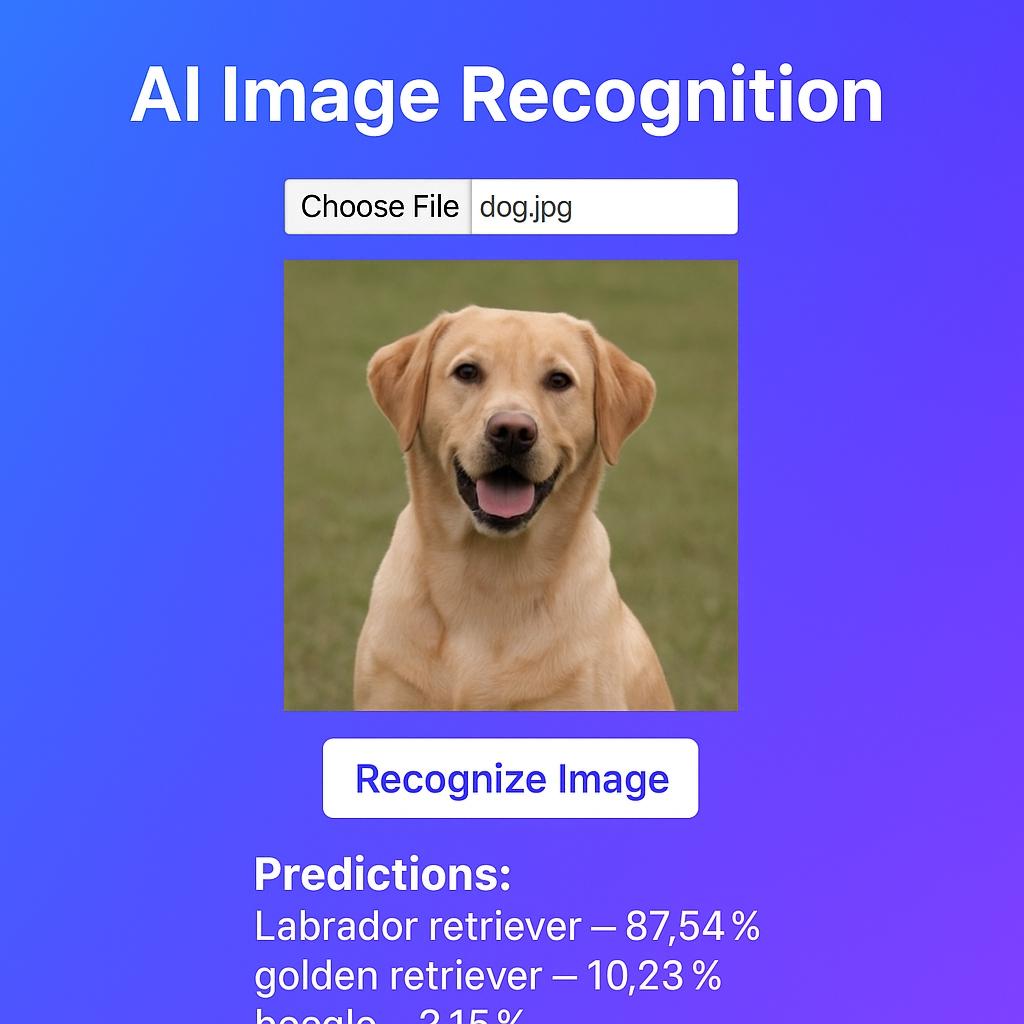
“react-dom”: “latest”,

“@tensorflow/tfjs”: “^4.12.0”,

“@tensorflow-models/mobilenet”: “^2.1.0”

}

}

Github link:<https://github.com/ShreenithiE/MVP-internship>

Versel link:<https://mvp-internship-theta.vercel.app/>