**Step 1: Sending Base64 Captcha to API Gateway**

To send the base64 encoded captcha to the API Gateway when the "Predict" button is clicked, you'll need to make an HTTP POST request from your website's JavaScript. You can use the **fetch** API or any JavaScript HTTP library for this.

1. In your HTML, make sure you have a button with an id for triggering the prediction:

<button id="predictButton">Predict</button>

Predict</button>

1. In your JavaScript file (let's call it **script.js**), add an event listener to the button:
2. document.getElementById("predictButton").addEventListener("click", async () => {
3. const base64Captcha = "your\_base64\_data\_here"; // Get the base64 captcha from your source
4. const apiUrl = "YOUR\_API\_GATEWAY\_URL\_HERE"; // Replace with your API Gateway URL
5. const response = await fetch(apiUrl, {
6. method: "POST",
7. body: JSON.stringify({ data: base64Captcha }),
8. headers: {
9. "Content-Type": "application/json"
10. }
11. });
12. const predictionResult = await response.json();
13. console.log("Prediction Result:", predictionResult);
14. });

});

**Step 2: Finish the Website HTML, CSS, and JS**

Since you're familiar with HTML, CSS, and JS, you can continue developing your website based on your design and layout preferences. You can style your captcha display and prediction button using CSS, and you can integrate the JavaScript code provided above into your **script.js** file. Additionally, you can add elements to display the prediction result on your webpage.

**Step 3: Varying Captchas Displayed on the Website**

To vary the captchas displayed on your website, you'll need to generate or obtain different captcha images and their corresponding base64 encodings. You can host these images on your server or use a third-party service to generate captchas.

1. Create a collection of captcha images and generate their base64 encodings. You can use various libraries like Pillow for generating captcha images in Python.
2. Store these base64-encoded captcha images in a database or a JSON file.
3. Modify your JavaScript code to randomly select a captcha image and its base64 encoding to display on your website.
4. const captchaData = [
5. { base64: "base64\_encoded\_image1", label: "ABCD" },
6. { base64: "base64\_encoded\_image2", label: "WXYZ" },
7. // Add more captcha data entries
8. ];
9. // Randomly select a captcha data entry
10. const randomCaptcha = captchaData[Math.floor(Math.random() \* captchaData.length)];
11. // Display the captcha image on your webpage
12. const captchaImageElement = document.getElementById("captchaImage");
13. captchaImageElement.src = randomCaptcha.base64;

Make sure to adjust the HTML structure and JavaScript logic to accommodate displaying the captcha image and receiving predictions.

With these steps, you'll have a functional webpage that displays captchas, sends them to your AWS Lambda function through API Gateway for prediction, and displays the prediction result. Remember to replace placeholders like **YOUR\_API\_GATEWAY\_URL\_HERE** and **your\_base64\_data\_here** with actual values.