RONIT JAIN

SRN - 202201393 | c3 - 74

Project 2

Apply Captcha on Noble Hospital which must introduce client-side input validation to improve the user interface and enhance security, guaranteeing data integrity by conducting comprehensive checks of format, length, and type validation within their web application



**HIGHLIGHT ON THE PROBLEM STATEMENT**

The earlier problem statement suggested that "Noble Hospital" needs to implement client-side input validation in its web application. Now it says to apply Captcha validation as well.

A CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) is a type of challenge-response test used in computing to determine whether the user is human or not. CAPTCHAs are typically used in web forms to prevent automated software (bots) from submitting the form, thereby reducing spam and abuse.

**THEORY**

In summary for the project-1, the problem statement emphasized on the importance of implementing client-side input validation in the web application of Noble Hospital. This implementation aimed to provide a better user experience, enhance security, and ensure the integrity of the data collected from users.

**IMPLEMENTATION OF CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart)**

An HTML block containing CAPTCHA form is embedded which by default disabled, It comes as a pop-up window specially for handling CAPTCHA validation.

CAPTCHA is disables until the form is submitted successfully. Every time you refresh a new CAPTCHA is generated.  
If a form is submitted successfully, but the CAPTCHA is failed the input of the form stays there, preventing it form refreshing.

<!—HTML CODE SNIPPET --!>

    <div id="id01" class="modal">

        <form class="modal-content animate" id="c-form">

            <span onclick="document.getElementById('id01').style.display='none'" class="close"

                title="Close Modal">&times;</span>

            <div class="container">

                <div class="captcha-img"><del><p class="special-elite-regular" id="captcha"></p></del></div>

                <div class="captcha-inp">

                    <input id="c-input" type="text" placeholder="Enter Captcha" name="uname" required>

                </div>

                <div class="captcha-submit"><button id="c-submit" type="submit">ENTER</button></div>

                <div id="c-msg"></div>

            </div>

        </form>

    </div>

<!—JS CODE SNIPPET --!>

function popUpCaptcha() {

*// Clear old input*

    document.getElementById("c-input").value = "";

*// Access the element to store the generated captcha*

    captcha = document.getElementById("captcha");

    let uniquechar = "";

    const randomchar = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789";

    let hasCapital = false;

    let hasSmall = false;

    let hasNumber = false;

    while (!(hasCapital && hasSmall && hasNumber)) {

*// Reset the string for each attempt*

        uniquechar = "";

        for (let i = 0; i < 5; i++) {

            const char = randomchar.charAt(Math.floor(Math.random() \* randomchar.length));

            uniquechar += char;

            if (!hasCapital && /[A-Z]/.test(char)) {

                hasCapital = true;

            }

            if (!hasSmall && /[a-z]/.test(char)) {

                hasSmall = true;

            }

            if (!hasNumber && /[0-9]/.test(char)) {

                hasNumber = true;

            }

        }

    }

*// Store generated input*

    captcha.innerHTML = uniquechar;

}

document.getElementById("c-form").addEventListener("submit",(e)=>{

    e.preventDefault();

    captcha = document.getElementById("captcha").innerHTML;

    const userInput = document.getElementById("c-input").value;

    if(userInput === captcha){

        alert("SUCCESSFUL");

        direct();

    }

    else{

        alert("Captcha doesn't match");

        document.getElementById('id01').style.display = 'none';

        popUpCaptcha();

    }

});

function direct(){

    window.location.href="HOME.html";

*//    window.location.href="https://www.jehangirhospital.com/";*

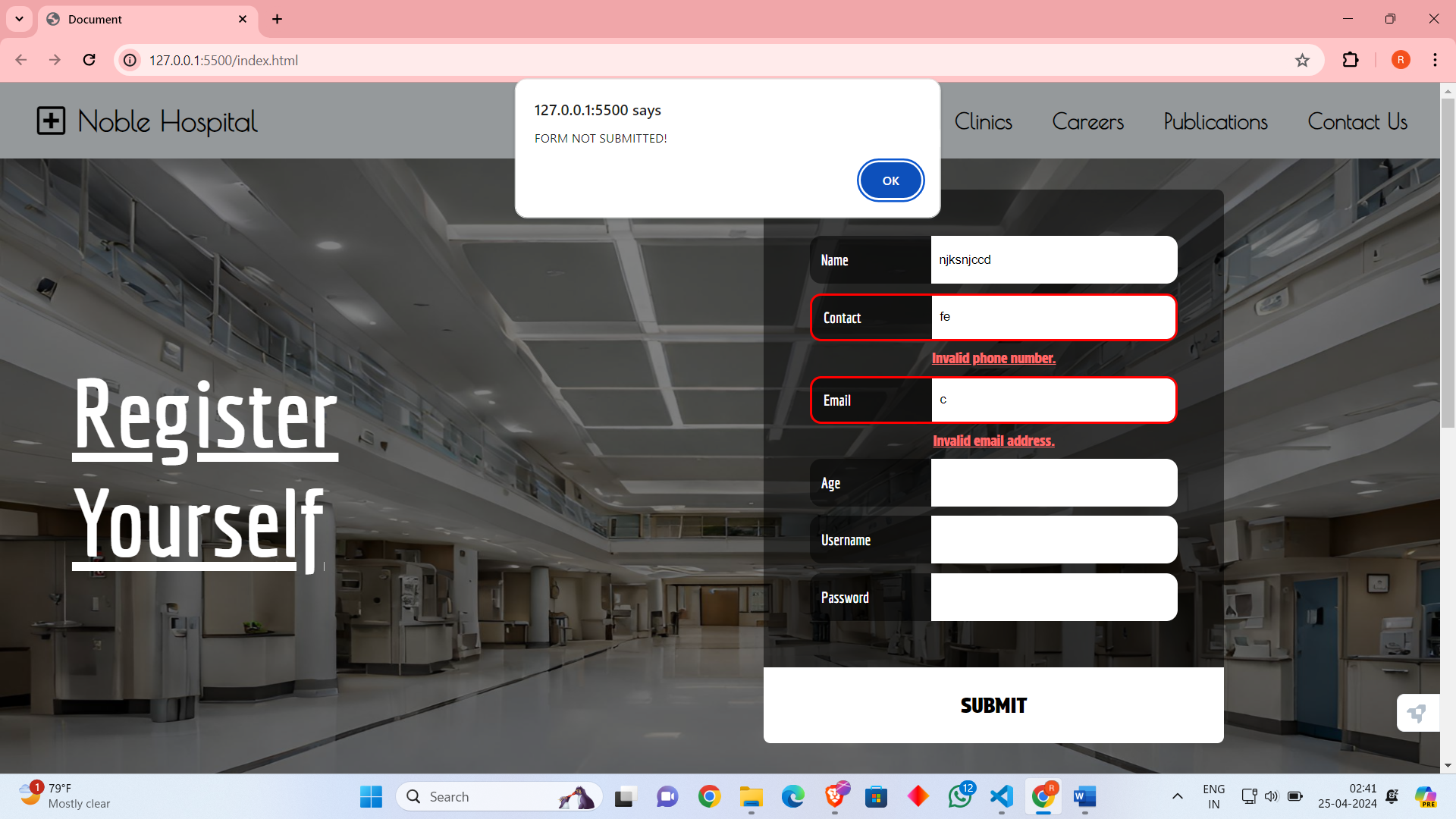
}

The “popUpCaptcha()” function generates a random string of length 5 containing at least one Capital letter, one small letter and a number.

The event listener is to validate the input of the CAPTCHA form, if the validation is successful the page gets redirected to “HOME.html”.

<!—OUTPUTs—>

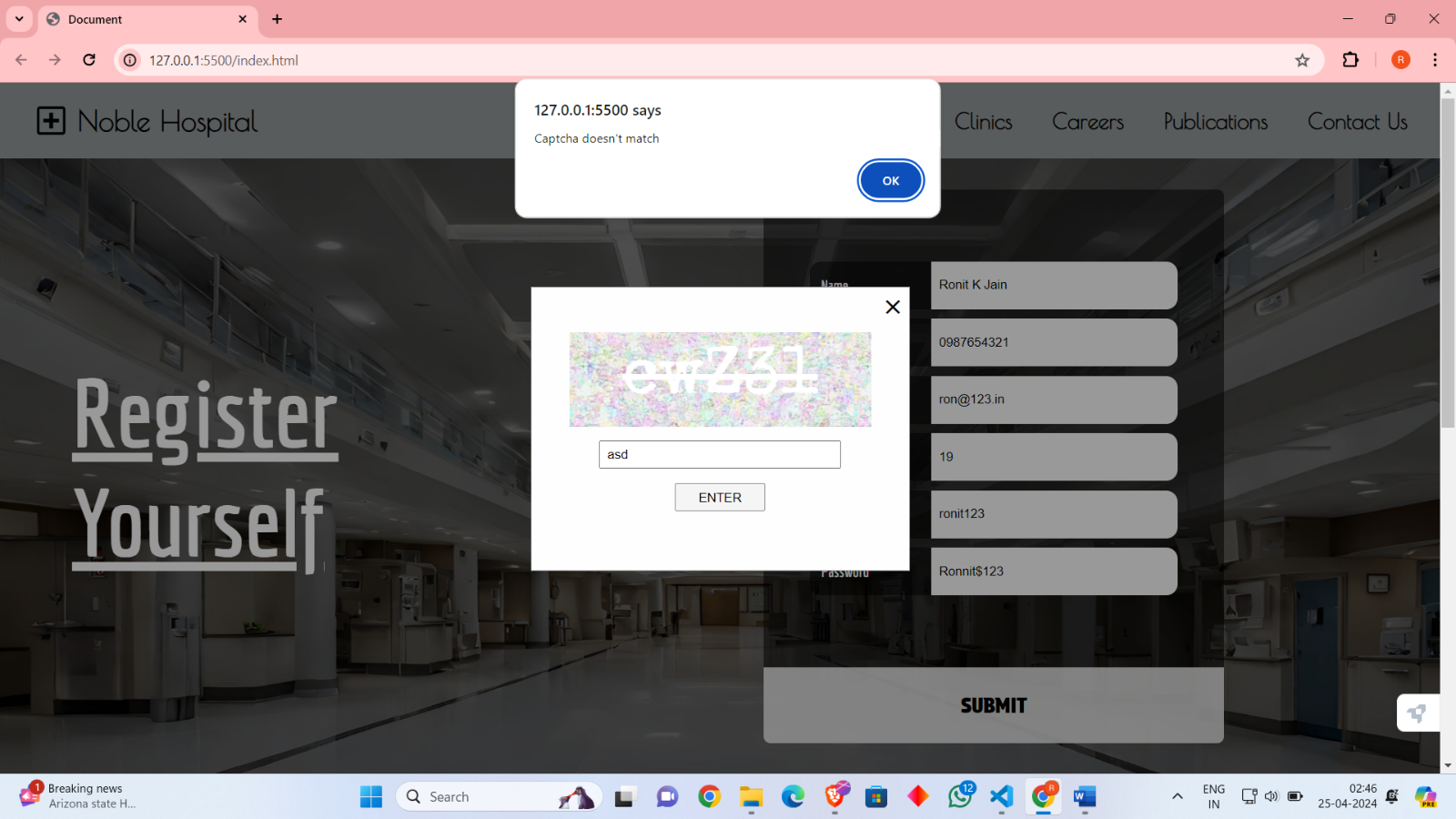
The CAPTCHA form is by default disabled and is never seen until the form is filled with zero error.



Once form is submitted successfully the CAPTCHA pop-up comes.

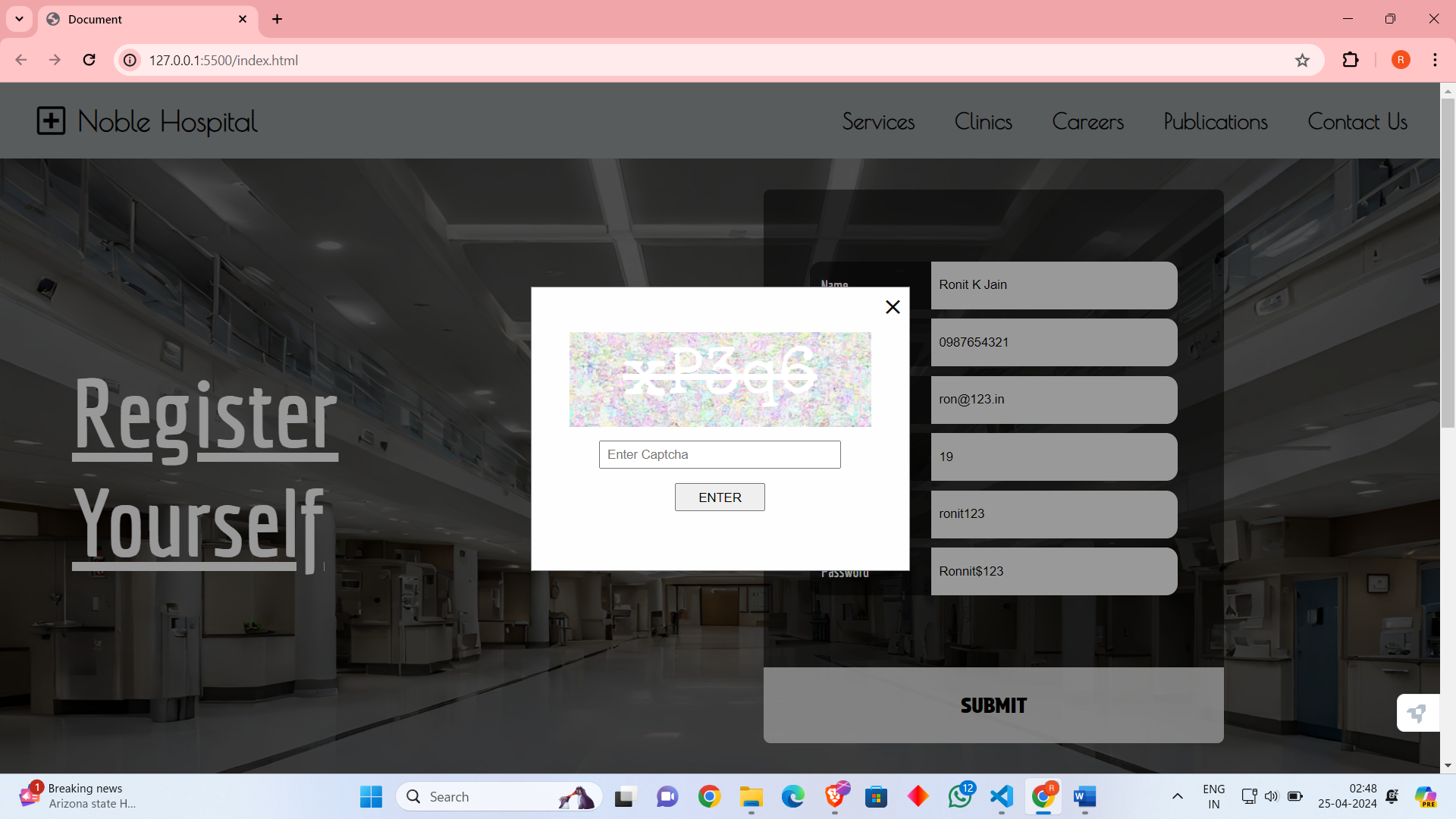


If the CAPTCHA validation fails the previous details are still stored.

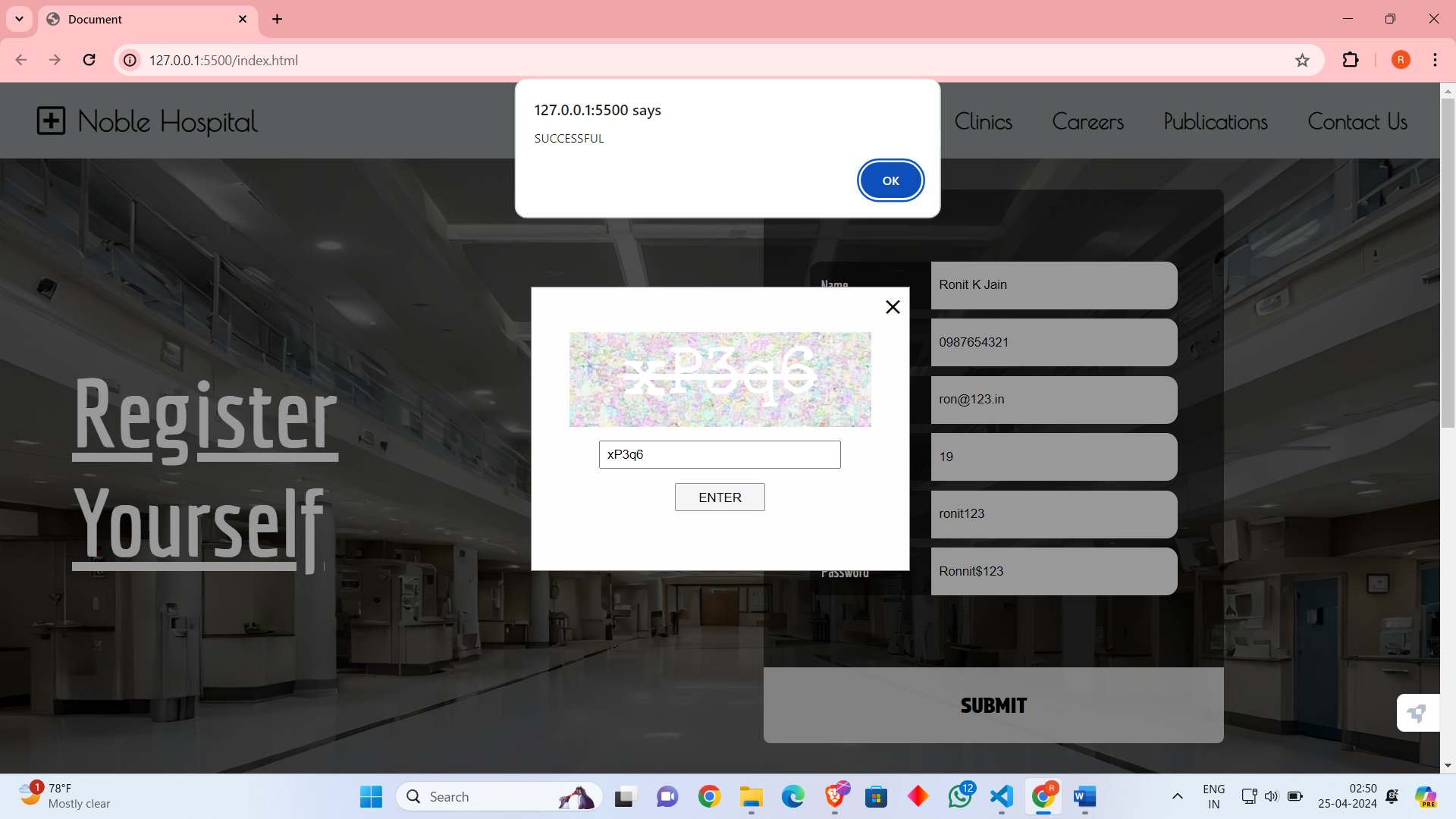


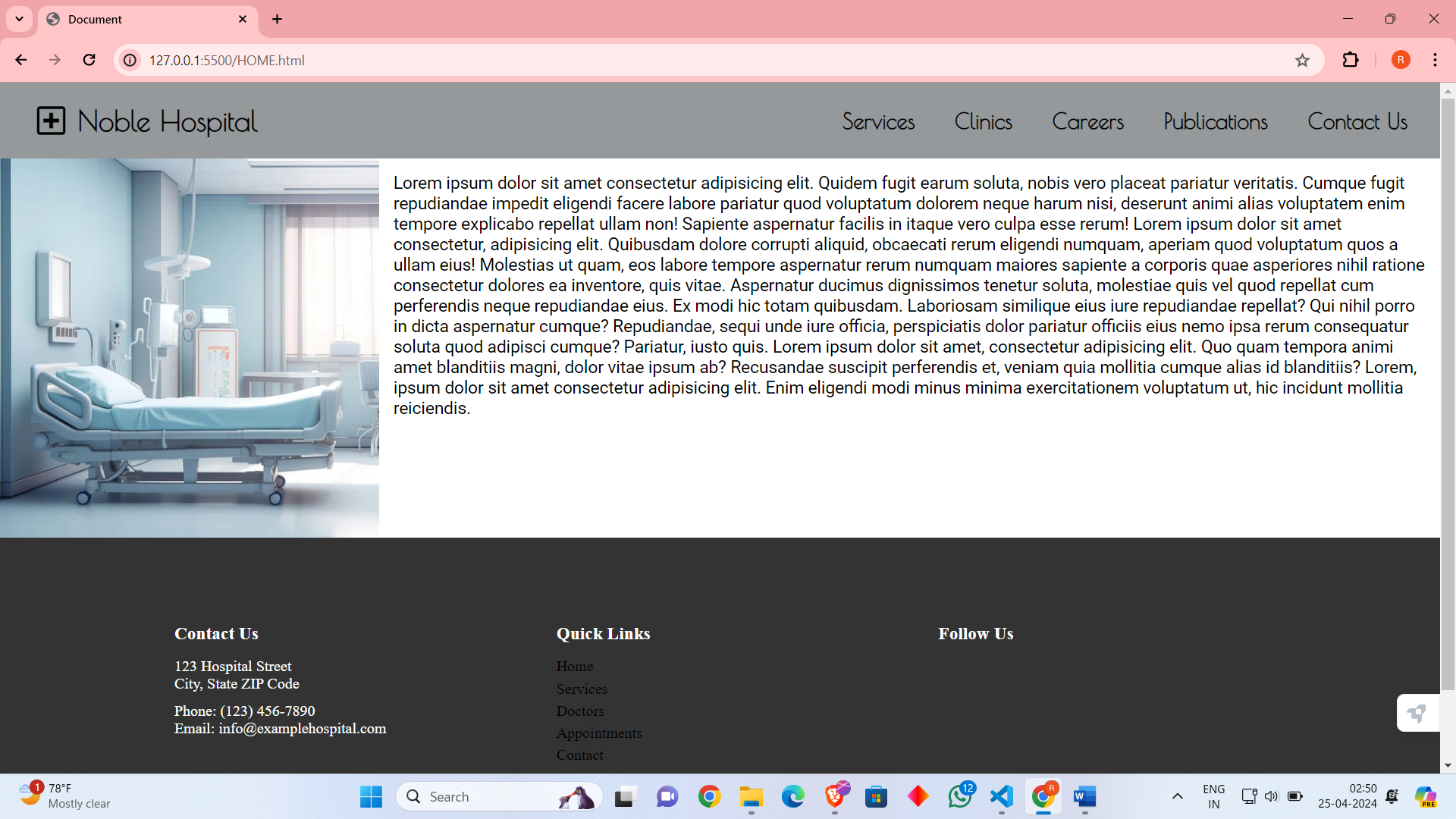


Every time a new CAPTCHA is generated.



If the CAPTCHA form is submitted successfully page is redirected to home,html





**CONCLUSION**

Embedding CAPTCHA in a form is a practical and effective way to enhance security and prevent automated abuse. By presenting a challenge that requires human-level cognition to solve, CAPTCHA helps verify that the user interacting with the form is a real person, not a bot. This reduces spam submissions, protects against brute force attacks, and enhances the overall security of the website. Including CAPTCHA in a form is a proactive measure that contributes to a better user experience by ensuring that legitimate users can interact with the website without interference from automated scripts or bots.