* 1. las pruebas que realizaría al sitio serian:

prueba smock para verificar a nivel general el funcionamiento de la aplicación y conocer cómo funciona la aplicación a nivel general para poder entrar a identificar y realizar los casos de prueba a nivel funcional

1. pruebas de enlaces para verificar los hipervínculos internos y externos de la aplicación los externos mas que todo para verificar que si consume otros service lo haga.
2. Probar los formularios de la aplicación es decir el tipo de dato que debe capturar cada caja que realice todas y cada una de las validaciones como correos tipos de datos, que realiza las acciones mas recurrentes como inserción , actualización , eliminación de registros o información
3. Probar cookies es decir verificar como funciona la aplicación con cookies deshabilitadas y habilitadas y comprobar que estén activas por un periodo de tiempo especificado
4. Validaciones de html o graficas en cuanto a sintaxis etcétera, adicional probar e formato xml o jeison dependiendo consumiendo los diferentes servicios que ofrece la aplicación para verificar respuestas , tiempo de respuesta aserciones , tiempos de respuesta excetera

Para ello con soap ui, postman dependiendo

Pruebas de compatibilidad con varios navegadores ya que hoy en día es común que un usuario use diversos tipos de estos exploradores

La mayoría de la cobertura de las pruebas se enfocaría en la parte funcional con los casos de prueba es bueno tener acceso a las bases de datos para identificar data y verificar que las operaciones actualicen y se reflejen correctamente en las tablas

* 1. defectos:

link de Tweet no funciona; severidad alta ya que no permite acceder al hipervínculo

share aparece un numero y al hacer on mouse aparece el nombre severidad baja ya que permite acceder al hipervínculo

comparte enlace con seguidores twiter ([LATEST] Best Resources to Learn Selenium Webdriver https://ultimateqa.com/best-selenium-webdriver-resources/?utm\_source=twitter&utm\_medium=Social&utm\_campaign=SocialWarfare vía @Nikolay\_A00)

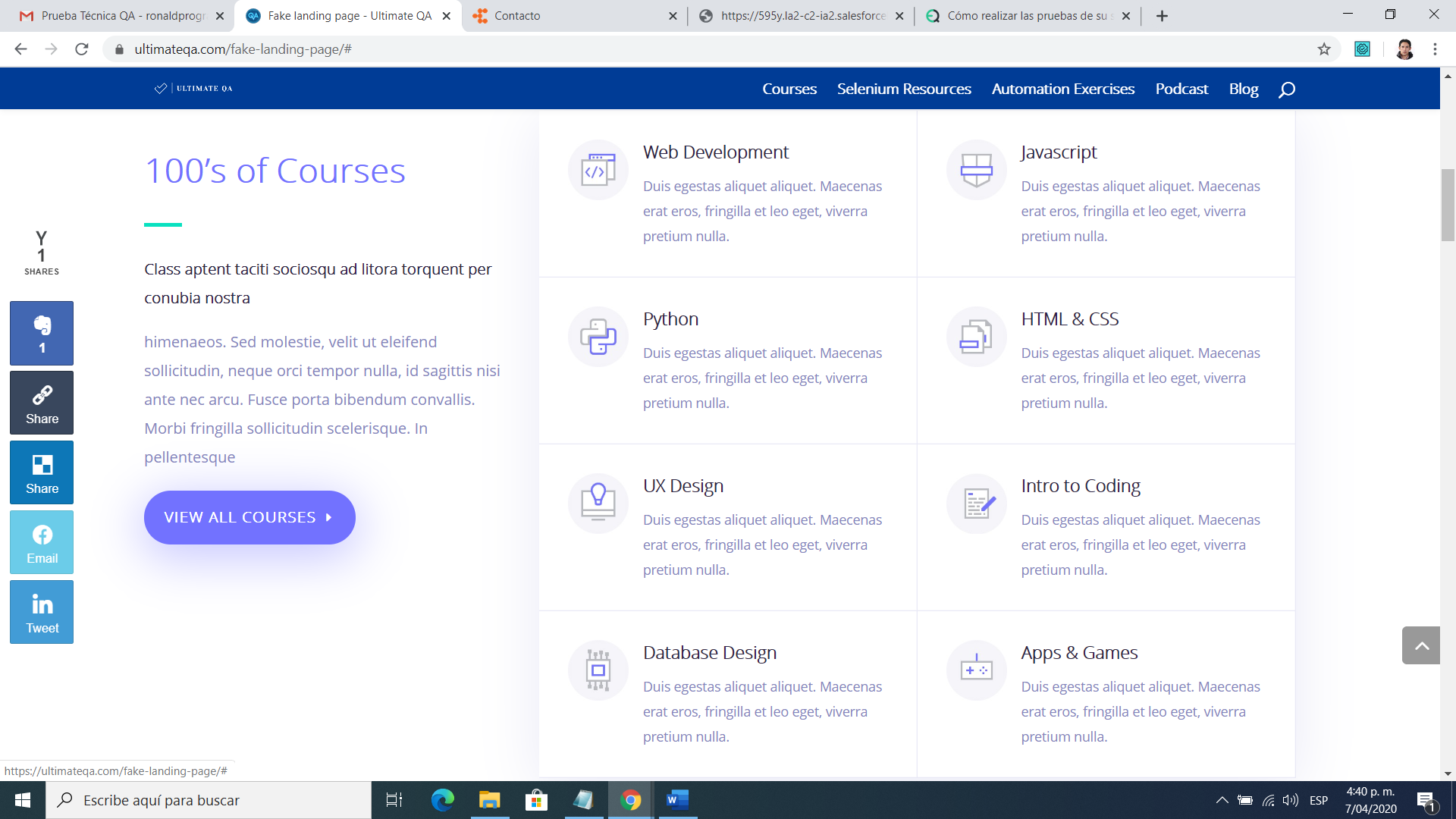
por seguridad y estética no debe aparecer el hipervínculo o url completo de acceso al sitio

ejercicios de automatización—pagina gran con muchos elementos--- se usa repetitivamente un botón que hace lo mismo no es funcional severidad baja

botón view courses no funcionan severidad alta ya que no permite ir al enlace especificado

botón view all couses direcciona a la pagina de inicio no muestra los cursos severidad alta

este menú de opciones direcciona a la pagina que los contiene no muestra información severidad alta o, funcionalmente no es viable n menú que direccione al mismo sitio



En el formulario <https://ultimateqa.com/filling-out-forms/>

Al enviar no valida el campo nombre y permite enviar datos numéricos severidad baja por que es agregar una validación

Al introducir en el campo Captcha total suma un carácter o letra debería indicar que solo admite valores numéricos no lo valida severidad baja agregar validación

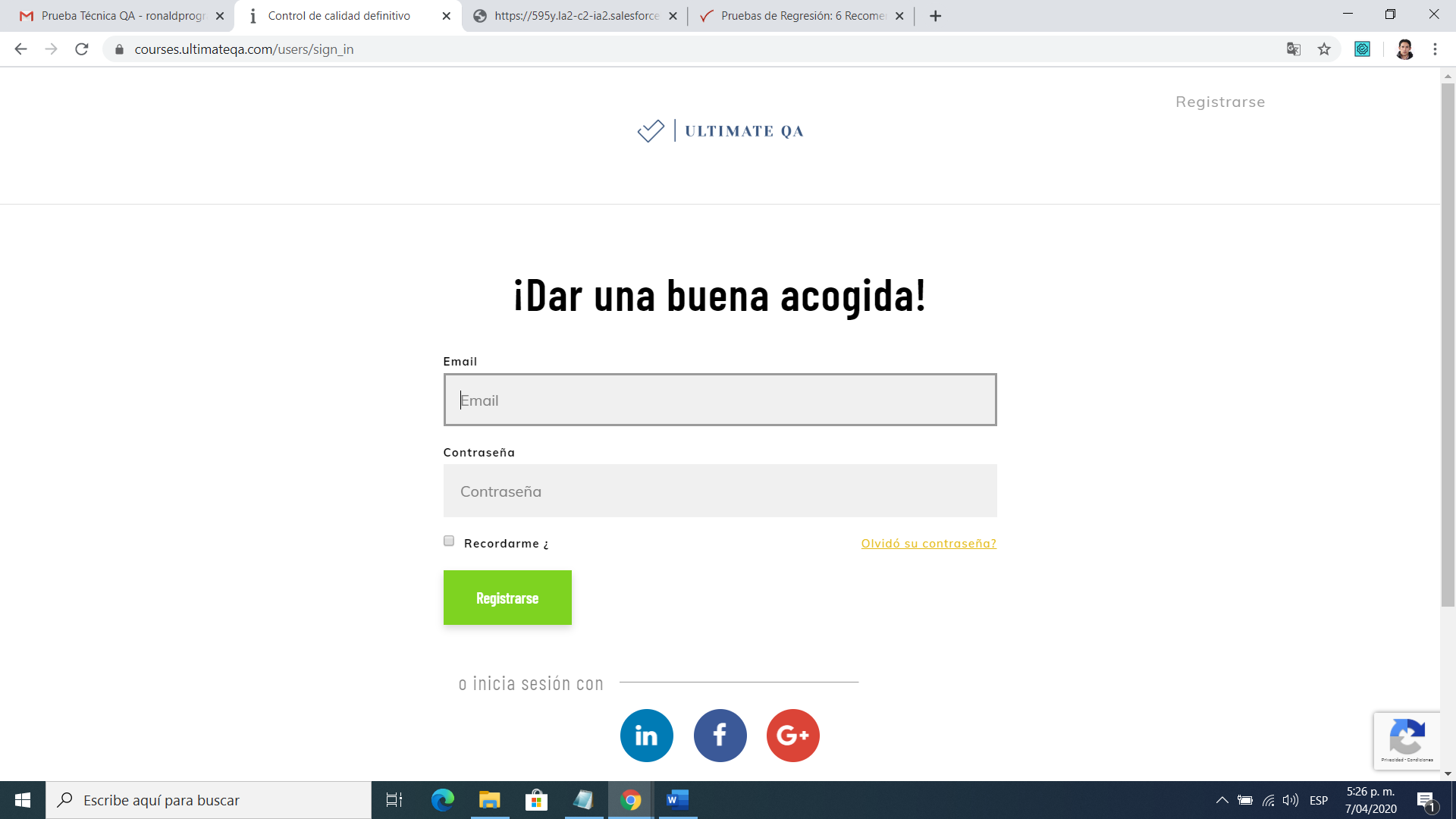
Al registrarse

En el link :

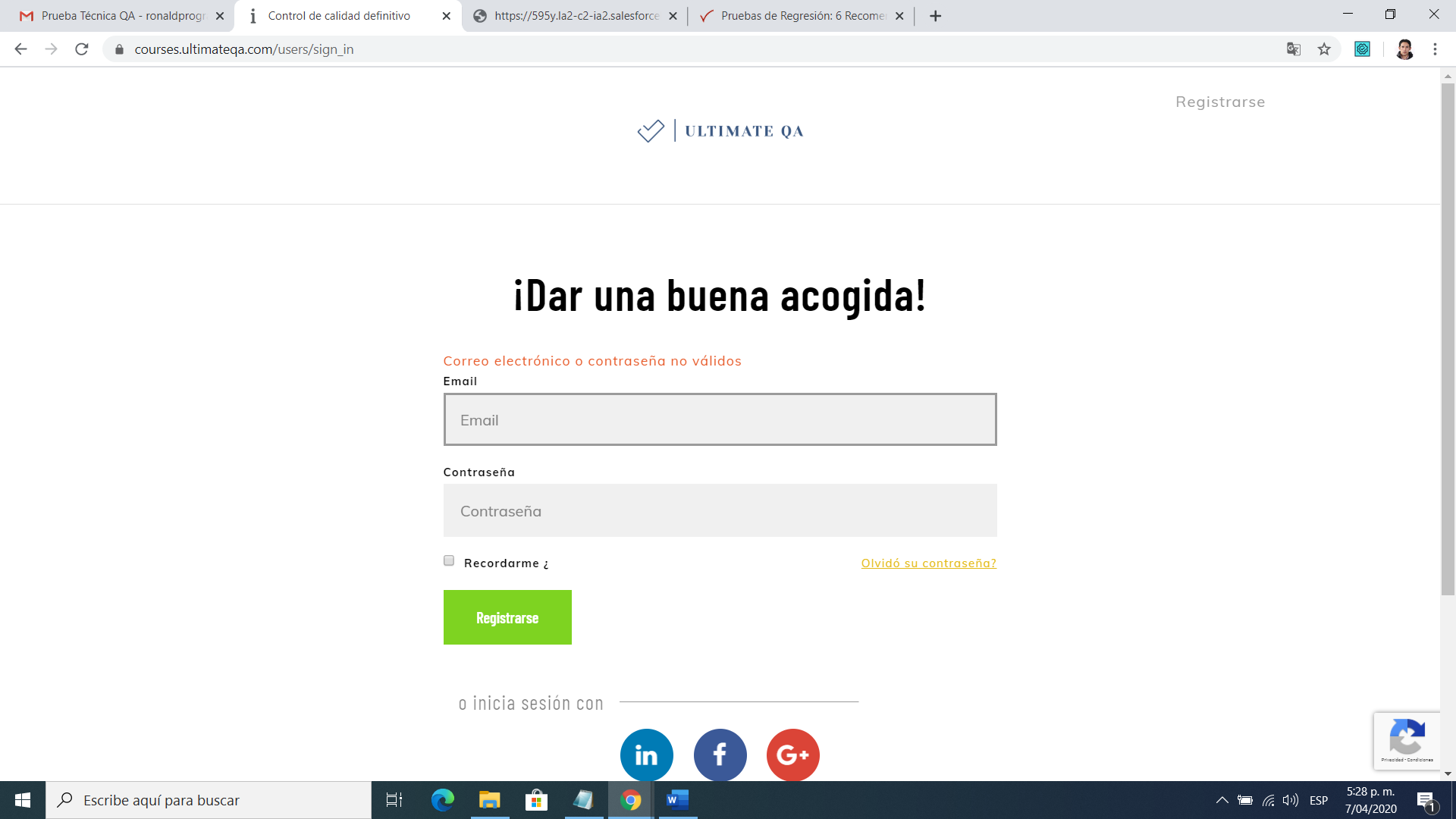
<https://patient-waterfall-9970.ck.page/a5bc75023d>

al realizar suscripción no valida que el correo ingresado exista indicando que el correo de notificación de inscripción fue enviado por lo general lo valida antes de enviar correo

en la opción cursos registrarse al ejecutar el evento Registrarse:



Sin ingresar correo o información me despliega el mensaje:



No me permite realizar registro con correos existentes y validos severidad alta porque no permite ingresar usuarios a la plataforma.

1.3 verificar catcha del formulario <https://ultimateqa.com/filling-out-forms/>

Introducir un numero erroneo en la suma , debe indicar Ha introducido un número equivocado en captcha.

Enviar en vacío el campo total suma debe indicar Por favor, complete los siguientes campos: Captcha

Introducir en el campo total suma un carácter debería indicar que solo admite valores numéricos

El caso exitoso es colocar en el capcha el total de la suma correcta y verificar que envíen el formulario exitoso

* 1. asumiendo que los errores fueron corregidos se entraría en una etapa no de retest especifico o puntual sobre el error si no plantear la estrategia sobre las pruebas de regresión acá puede ser útil la automatización ya que puede identificar errores en lo que ya se había probado y evita volver a hacer pruebas manuales sobre modules adicionales a los cuales se integra el modulo o ampliación sobre el cual nos hicieron la regresión en el área de desarrollo .

identificar rápidamente cuáles deberían ser los casos más importantes a probar de acuerdo a la severidad de los bug y la integración

diseñar menor numero de casos como sea posible tratando de obtener la mayor cobertura de todos los módulos involucrados para reducir tiempos de ejecución

identificar las rutas criticas y sus jerarquías o árbol para poder probar puntual y rápidamente la transacción

identificar los módulos que específicamente pueden verse afectados en la regresión con el fin de priorizar las pruebas en estos, y realizar sobre el resto del sistema unas pruebas de humo ágilmente

contemplar un plan de automatización para tener script que pueden ejecutarse varias veces evidenciando como funcionaba como funciona con la regresión y como debe funcionar la aplicación reduciendo desgaste del equipo y evidenciándolo de forma transparente en caso de detectar fallos