NOMBRE: Jonathan Rios Gómez - 1036654523

**Caso de negocio**:

En la página de Choucair Testing, se requiere realizar las pruebas de la sección de “Empleos”. Las pruebas deben verificar diferentes atributos de calidad.

Usted es responsable de las pruebas por ende le piden que las realice. A continuación, se muestran los criterios a cumplir:

Ruta de la aplicación: <https://www.choucairtesting.com/>

1. **Criterios tácticos:**
2. Determine que debe probar (Para esto puede explorar la sección como guste y parta de la condición de que existen errores) para asegurar que se cumple con el mínimo de calidad.
3. Considere la creación de casos de prueba. (Considerar pasos si es necesario)
4. De estos casos, clasifique que casos son automatizables y no automatizables.
5. Indique si existen y cuáles son las precondiciones para ejecutar los casos de prueba.
6. Presente estadísticas de prueba que permitan describir la conclusión de un indicador que defina.
7. Sugiera mejoras al módulo probado.
8. **Criterios técnicos:**
9. Realice la automatización (programación) de los casos de prueba que seleccionó como automatizables. Se requiere implementar el uso de las siguientes Herramientas:
   * + Lenguaje: Java
     + Tipo de proyecto: Deseable Gradle
     + Framework de ejecución: Junit
     + Framework de pruebas: Serenity/Selenium
     + Driver o navegador: Chrome
     + Arquetipo: Page Object Model o Screenplay
     + Escribir HU en: Cucumber
     + Ejecución: Junit o Jenkins Local
10. Realice la ejecución de lo automatizado, con sus respectivas evidencias.
11. Colocar el código automatizado en **github** y enviarlo. (Repositorio público)
12. **Agilismo**
13. De las sugerencias de mejora propuestas, defina historias de usuario que plantearía para un nuevo desarrollo, con sus respectivos criterios de aceptación.
14. Teniendo en cuenta que estas historias de usuario se seleccionaron en un sprint, liste que ***tareas de prueba y de desarrollo*** son necesarias realizar estas historias.
15. **Conceptos Programación / CI / CD**

Describa con sus propias palabras cada uno de los siguientes conceptos:

1. Principios SOLID
2. Patrón Singleton
3. Patrón FIRST
4. Patrón AAA
5. Pull Request
6. Release Train
7. Quality Gates
8. Diferencias servicios SOAP / REST

¡Este es un reto! Por ende, puede consultar toda la información que sea necesaria u otra opción es preguntar las dudas que tenga a personas conocidas.

**Plan de pruebas**

**Objetivo de plan de pruebas**

Establecer las condiciones en el tiempo determinado para la aplicación de las pruebas, con el fin de obtener un modelo el cual pueda comprender la totalidad de las funcionalidades requeridas para el correcto funcionamiento de la plataforma de Choucair, todo esto dentro del alcance admitido para el caso de uso y cumplir con las expectativas esperadas por el cliente.

**Alcance de las pruebas**

Para el caso de uso en cuestión se probarán atributos tanto funcionales, como de seguridad, intuitivos y de robustez, todo esto se deberá hacer desde el rol de usuario, probando acciones únicamente para el perfil de usuario en el apartado de empleo, que se encuentra en la barra de navegación superior de la página, dichas acciones incluyen ir al portal de empleo, verificar el redireccionamiento al formulario y el registro de la información solicitada.

**Estrategia de pruebas**

Una vez se encuentren todos los componentes del sistema definidos en el alcance, se realizarán las pruebas en donde el responsable de alguna o algunos componentes específicos del sistema, debe verificar que se cumpla la calidad esperada por parte del cliente, además debe tener en cuenta las prioridades establecidas dentro de los requerimientos al momento de ejecutar dichas pruebas y hacer los reportes correspondientes de cada una de ellas.

Pruebas Funcionales:

* Se verificará que se carguen completamente las opciones de ¿Qué es Choucair? y prepararse para aplicar.
* Se verificará que, al ser pulsada las dos opciones anteriores, el sistema ubique correctamente al usuario correctamente dentro del punto especifico de la página.
* Evaluar que al pulsar el botón “Ir al portal de empleos” la página permite realizar correctamente la acción de cambiar de sitio y preguntará al usuario si desea continuar.
* Probar el botón “Continuar” el cual deberá ingresar al sistema magneto.
* Evaluar que al ser pulsada la opción “Regístrate”, se mostrará la información que deberá diligenciar el usuario.
* Verificar que funciona la opción “Deseo llenar manualmente mis datos” al aceptar los términos y condiciones.
* Verificar que se habilita el botón “Continuar” al aceptar los términos y condiciones.
* Verificar la obligatoriedad de campos.

Pruebas para atributos intuitivos:

● Se evaluará si realmente aparecen los mensajes informativos de error para avisar al usuario cuando algo vaya mal en la pantalla de la página.

● Verificar que el sistema enviará mensajes de validación al usuario en la pantalla de registro de datos.

Pruebas de robustez:

● Se verificará que todos los campos obligatorios dentro de la pantalla de Registrate, permitan el envío del formulario solamente si están correctamente diligenciados, así mismo que lo impidan si falta algún campo obligatorio en dicho formulario.

**Elementos a no probar**

En el alcance de las pruebas no están incluidas las pruebas unitarias, de regresión y de componentes, debido a que el desarrollo del código fuente es implementado por una otra área no se puede acceder a él.

Tampoco se pueden realizar validaciones sobre la base de datos, al igual que en el punto anterior la base de datos del sistema no ha sido entregada para la fase de pruebas y no hay forma de acceder a ella para evaluar los criterios involucrados.

No sé va a probar la funcionalidad en un dispositivo móvil o en un navegador diferente a Google Chrome, tampoco se tendrá en cuenta la opción de subir la hoja de vida.

**Criterios de aceptación**

* Todos los casos de prioridad de alto riesgo deben ser probados y ser exitosos en un 100% y como mínimo un 95% en los casos de prioridades medias y bajas con el fin de garantizar la mayor calidad posible del sistema.
* Garantizar el uso debido de todos los casos acorde a los requerimientos establecidos por el cliente y el alcance del proyecto.
* Todos los casos de prueba definidos dentro del alcance debieron ser evaluados en lo acordado, además de entregar y tener la respectiva documentación de la funcionalidad del sistema.

**Entregables**

-   Documentación del plan de pruebas.

-   Casos de pruebas implementados.

-   Documentación y reportes de los casos de pruebas.

- Códigoutilizado para la automatización.

**Riesgos**

● Dependencias con Desarrollos.

● Dependencias con otros proyectos.

● Disponibilidad de recursos.

● Restricciones de tiempo.

● Premisas que resultan no ser ciertas. utilizado en la automatización.

**Glosario**

* Caso de prueba: es la unidad más pequeña del plan de prueba, que incluye una descripción de las acciones y los parámetros necesarios para lograr y verificar el comportamiento esperado de una función en particular o la parte del software probado.
* Herramienta de testing: software especializado en la comprobación y en la automatización de algunos procesos, estas pueden ser herramientas de gestión de pruebas, herramientas para pruebas funcionales, herramientas para pruebas de carga y rendimiento.

Las siguientes definiciones fueron extraídas de (Cillero manuel, 2009).

* Pruebas unitarias: son una forma de comprobar que un fragmento de código funciona correctamente, se hacen sin necesidad de manuales y pueden repetirse indefinidamente las veces que se requieran.
* Pruebas de regresión: la prueba de regresión consiste en probar un sistema que ha sido analizado previamente para asegurar que no se haya introducido algún tipo de defecto como resultado de cambios realizados.
* Pruebas de prototipo en interfaz: Pruebas realizadas a los prototipos para verificar que las acciones o su funcionamiento sea el indicado.
* Pruebas de usabilidad: son procedimientos con los que se puede probar la usabilidad de una web, probando si las formas de encontrar información en el sitio son las más eficientes para cualquier tipo de usuario.
* Pruebas de información y simulación: tienen como objetivo ejercitar profundamente el sistema comprobando la integración del sistema de información globalmente, verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen y con el resto de sistemas de información con los que se comunica.

Caso de prueba #1 Registro.

|  |  |
| --- | --- |
| **Objetivos de la prueba** | Verificar que el usuario puede ingresar al portal de empleos e ingresar al formulario sus datos personales. |
| **Técnicas** | * Ingresar al sitio web oficial y pulsar la opción de “Empleo” en el menú superior. * Una vez ubicado en esta pestaña pulsar el botón “Ir al portal de empleos”. * Seleccionar la opción de “Deseo llenar manualmente mis datos”. * Dar clic en continuar aceptando términos y condiciones. * Diligenciar correctamente todos los campos obligatorios. |
| **Interfaz asociada** |  |
| **Resultados** | * El sistema permitió el ingreso de diferentes usuarios probados, cuando los datos ingresados fueron correctos y muestra el mensaje de error correctamente cuando algún campo es incorrecto. |

Casos de prueba #2 Navegación.

|  |  |
| --- | --- |
| **Objetivos de la prueba** | Verificar que el usuario puede dirigirse a los apartados “¿Qué es Choucair?” y “Prepararse para aplicar” |
| **Técnicas** | * Pulsar el botón ¿Qué es Choucair? * Pulsar el botón de prepararse para aplicar. * En para ambos casos, verificar que el sistema muestre la interfaz correspondiente. |
| **Interfaz asociada** |  |
| **Resultados** | El sistema permitió la ubicación en un punto específico dentro de la pestaña al pulsar ambas opciones. |

1. **Conceptos Programación / CI / CD**

Describa con sus propias palabras cada uno de los siguientes conceptos:

1. Principios SOLID: Es un acrónimo que representa 5 principios claves en el desarrollo de software, los cuales definen un modelo de trabajo para ordenar funciones, métodos y estructura de clases.
2. Patrón Singleton: Restringe la creación de objetos, para que este solo se pueda instanciar una única vez.
3. Patrón FIRST: Son características que deben tener las pruebas unitarias para mejorar la calidad.
4. Patrón AAA: Este patrón permite organizar las pruebas unitarias al dividirla en 3 sesiones.
5. Pull Request: Es una petición que se le hace al propietario de un repositorio para incorporar commits.
6. Release Train: Es un método que utilizan los equipos agiles para planificar la entrega de software.
7. Quality Gates: Permite mejorar la calidad de la información de las organizaciones a través de métricas para la seguridad y vulnerabilidades-ç.

Diferencias servicios SOAP / REST

SOAP es parecido a un protocolo de comunicación en formato XML, utilizando protocolo HTTP, SMTP entre otros mientras que RESt se centran más en obtener peticiones sobre una URI en formato HTTP.