

Informe de pruebas manuales

REGISTRO DIGITAL DEL APRENDIZ (REDA)



1828182
JUNIO DE 2020

Servicio Nacional de Aprendizaje



Análisis y Desarrollo de Sistemas de Información

Elaborado por:

Breyder Camilo Gonzalez Castillo

María Camila Parra Bedoya

Santiago Jiménez Jiménez

Competencia: Calidad de software

Instructor:

Edilfredo Pineda Flórez

Medellín, Antioquia

Junio de 2020

Tabla de contenidos

	pág.
1. Introducción.....	3
1.1 Objetivos	3
1.2 Alcances	3
2. <u>Pruebas manuales (caja blanca)</u>.....	4
2.1 Caso de pruebas – Cambiar contraseña.....	4
2.2 Caso de pruebas – Registrar asistencia	6
2.3 Caso de pruebas – Buscar una ficha	7
2.4 Caso de pruebas – Iniciar sesión	9
2.5 Caso de pruebas – Agregar un nuevo usuario.....	11
2.6 Caso de pruebas – Registrar contraseña.....	14
3. Pruebas manuales (caja negra).....	16
3.1 Caso de prueba – Registrar un nuevo usuario.....	16
4. Conclusiones.....	20

1. Introducción

En este documento se dará información de las pruebas manuales que se realizaron a algunos módulos de REDA (Registro Digital del Aprendiz). Este, es un sistema de información que les permite a los instructores del Centro Textil y de Gestión Industrial registrar la asistencia de los aprendices por medio de la huella dactilar usando un lector biométrico.

Con estas pruebas se comprueba el total funcionamiento de este sistema, estas se hicieron con una herramienta llamada Selenium IDE que comprueba el funcionamiento del sistema por medio de unas acciones determinadas.

1.1 Objetivos del sistema

REDA (Registro Digital del Aprendiz) fue creado para registrar la asistencia de los aprendices de forma ágil y automatizada en base a la iniciativa Cero papel en el Sena, que plantea un apoyo al medio ambiente. Los propósitos generales de REDA son:

- Reducir o anular el uso de recursos como papel, tinta y energía.
- Sistematizar y dinamizar el proceso de registro de asistencia.
- Tener un mayor control de la asistencia de los aprendices.
- Informar a los aprendices de sus ausencias y recordarles la importancia de la gestión de las excusas.

1.2 Alcances.

REDA plantea ser un sistema de información que pueda ser escalable a nivel tecnológico, ofreciendo así nuevas posibilidades u opciones de uso que faciliten la tarea de tomar asistencia.

2. Pruebas manuales

2.1 Caso de pruebas - Cambiar Contraseña

No	01
Nombre/Identificador	CP_CambiarContraseña1
Descripción	Cambiar contraseña dejando todos los campos vacíos
Precondiciones	Debe tener una contraseña guardada en la base de datos.
Entradas	Contraseña anterior y contraseña nueva
Pasos	1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en "Cambiar contraseña". 3. Sin llenar ningún campo dar clic en "Guardar cambios".
Resultados esperados	Mostrará un mensaje que resaltará el campo "Contraseña actual" y no cambiará la contraseña.
Post condición	Se mantiene la misma contraseña en la base de datos.

No	02
Nombre/Identificador	CP_CambiarContraseña2
Descripción	Cambiar contraseña llenando todos los campos.
Precondiciones	Debe tener una contraseña guardada en la base de datos
Entradas	Contraseña anterior y contraseña nueva
Pasos	1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en "Cambiar contraseña". 3. Llenar todos los campos y dar clic en "Guardar cambios".
Resultados esperados	La pestaña recargará y se mostrará un mensaje indicando que la contraseña ha sido cambiada con éxito.
Post condición	La contraseña es cambiada y actualizada en la base de datos

No	03
Nombre/Identificador	CP_CambiarContraseña3
Descripción	Cambiar contraseña llenando solo un campo.
Precondiciones	Debe tener una contraseña guardada en la base de datos
Entradas	Contraseña anterior y contraseña nueva
Pasos	1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en "Cambiar contraseña". 3. Sin llenar ningún campo dar clic en "Guardar cambio".

Resultados esperados	Mostrará un mensaje que resaltará el campo "Contraseña actual" y no cambiará la contraseña.
Post condición	Se mantiene la misma contraseña en la base de datos.

No	04
Nombre/Identificador	CP_CambiarContraseña4
Descripción	Cambiar contraseña digitando datos incorrectos.
Precondiciones	Debe tener una contraseña guardada en la base de datos
Entradas	Contraseña anterior y contraseña nueva
Pasos	<ol style="list-style-type: none"> 1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en "Cambiar contraseña". 3. Digitar una contraseña actual incorrecta y dar clic en "Guardar cambios".
Resultados esperados	Mostrará un mensaje que dirá que la contraseña actual es incorrecta.
Post condición	Se mantiene la misma contraseña en la base de datos.

2.2 Caso de pruebas - Registrar asistencia

No	05
Nombre/Identificador	CP_RegistrarAsistencia1
Descripción	Iniciar el registro de asistencia sin llenar ningún campo.
Precondiciones	Debe existir una ficha creada en la base de datos.
Entradas	Número de ficha y fecha.
Pasos	1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en "Registrar la asistencia". 3. Sin llenar ningún campo, dar clic en "Comenzar registro".
Resultados esperados	Mostrará un mensaje que resaltará el campo "ficha" y no iniciará el registro.
Post condición	No se guarda el registro en la base de datos.

No	06
Nombre/Identificador	CP_RegistrarAsistencia2
Descripción	Iniciar el registro de asistencia llenando todos los campos.
Precondiciones	Debe existir una ficha creada en la base de datos
Entradas	Número de ficha y fecha
Pasos	1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en "Registrar la asistencia". 3. Llenar todos los campos y dar clic en "Comenzar registro".
Resultados esperados	Se iniciará el registro, guardándolo en la base de datos y redirigiendo al usuario a la pestaña donde se toma la asistencia de los aprendices.
Post condición	Se guardan todos los datos del registro en la base de datos.

No	07
Nombre/Identificador	CP_RegistrarAsistencia3
Descripción	Iniciar el registro de asistencia ingresando datos incorrectos.
Precondiciones	Debe existir una ficha creada en la base de datos
Entradas	Número de ficha y fecha
Pasos	1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en "Registrar la asistencia". 3. Llenando datos incorrectos y dar clic en "Comenzar registro".

Resultados esperados	Mostrará un mensaje que dirá datos incorrectos y no iniciará el registro.
Post condición	No se guarda el registro en la base de datos.

2.3 Caso de pruebas - Buscar ficha

No	09
Nombre/Identificador	CP_BuscarFicha1
Descripción	Buscar los registros de asistencia por número de ficha ingresando un número de ficha registrado en el sistema.
Precondiciones	Debe existir una ficha creada en la base de datos.
Entradas	Número de ficha
Pasos	<ol style="list-style-type: none"> 1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en el buscador de ficha. 3. Ingresar número de ficha. 4. Dar clic en la lupa u oprimir la tecla ENTER del teclado.
Resultados esperados	Redirigirá a una página que mostrará el resultado de búsqueda.
Post condición	No se altera nada en la base de datos.

No	10
Nombre/Identificador	CP_BuscarFicha2
Descripción	Buscar los registros de asistencia por número de ficha ingresando un número de ficha NO registrado en el sistema.
Precondiciones	Debe existir una ficha creada en la base de datos.
Entradas	Número de ficha
Pasos	<ol style="list-style-type: none"> 1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en el buscador de ficha. 3. Ingresar un número de ficha no existente en el sistema. 4. Dar clic en la lupa u oprimir la tecla Enter del teclado.
Resultados esperados	El usuario será redirigido a la pestaña de resultados, más no habrá información de una ficha a mostrar.
Post condición	No se altera nada en la base de datos

No	11
Nombre/Identificador	CP_BuscarFicha3
Descripción	Buscar los registros de asistencia por número de ficha sin llenar el campo.
Precondiciones	Debe existir una ficha creada en la base de datos.
Entradas	Número de ficha

Pasos	1. Ingresar usuario y contraseña y autenticarse como miembro del sistema. 2. Dar clic en la lupa u oprimir la tecla Enter del teclado.
Resultados esperados	Mostrará un mensaje que resaltará el campo "ficha".
Post condición	No se altera nada en la base de datos

2.4 Caso de pruebas - Iniciar Sesión

No	12
Nombre/Identificador	CP_iniciarLogin1
Descripción	Acceder al sistema con los datos de los campos correctos.
Precondiciones	El usuario ya debe estar creado por el administrador.
Entradas	Datos del usuario que quiere acceder al sistema.
Pasos	<ol style="list-style-type: none"> 1. Ingresar a la página web REDA. 2. Dar clic en el botón Iniciar sesión. 3. Llenar correctamente los campos requeridos. 4. Dar clic en el botón ingresar.
Resultados esperados	Ingresar correctamente al sistema.
Post condición	El usuario inició sesión para poder acceder al sistema.

No	13
Nombre/Identificador	CP_iniciarLogin2
Descripción	Acceder al sistema con los campos vacíos.
Precondiciones	El usuario ya debe estar creado por el administrador.
Entradas	Datos del usuario que quiere acceder al sistema.
Pasos	<ol style="list-style-type: none"> 1. Ingresar a la página web REDA. 2. Dar clic en el botón Iniciar sesión. 3. Dejar los campos requeridos vacíos. 4. Dar clic en el botón ingresar.
Resultados esperados	Se muestra un mensaje que indica "Los campos están vacíos".
Post condición	El usuario no podrá ingresar al sistema.

No	14
Nombre/Identificador	CP_iniciarLogin3
Descripción	Acceder al sistema con los datos de los campos incorrectos.
Precondiciones	El usuario ya debe estar creado por el administrador.
Entradas	Datos del usuario que quiere acceder al sistema.
Pasos	<ol style="list-style-type: none"> 1. Ingresar a la página web REDA. 2. Dar clic en el botón Iniciar sesión. 3. Llenar incorrectamente los campos requeridos. 4. Dar clic en el botón ingresar.
Resultados esperados	Se muestra un mensaje que indica "Usuario o contraseña incorrectas o no coincidentes con el cargo seleccionado".

Post condición	El usuario no podrá ingresar al sistema.
-----------------------	--

No	15
Nombre/Identificador	CP_iniciarLogin4
Descripción	Acceder al sistema con algunos campos vacíos.
Precondiciones	El usuario ya debe estar creado por el administrador.
Entradas	Datos del usuario que quiere acceder al sistema.
Pasos	<ol style="list-style-type: none"> 1. Ingresar a la página web REDA. 2. Dar clic en el botón Iniciar sesión. 3. Llenar algunos campos requeridos y dejar otros vacíos. 4. Dar clic en el botón ingresar.
Resultados esperados	Se muestra un mensaje que indica "Llenar los campos vacíos".
Post condición	El usuario no podrá ingresar al sistema.

2.5 Caso de pruebas - Agregar un nuevo usuario

No	16
Nombre/Identificador	CP_agregar1
Descripción	Agregar a un usuario llenando todos los campos.
Precondiciones	El administrador debe iniciar sesión para poder agregar a los demás usuarios.
Entradas	Datos del nuevo usuario que se creará.
Pasos	1. Ingresar al sistema con los campos que piden, cargo, documento y contraseña. 2. Darle clic en la opción agregar usuarios. 3. Llenar los campos requeridos por el sistema. correctamente, dar clic en enviar y en ese momento se agregará un usuario.
Resultados esperados	Se muestra un mensaje que indica que el registro ha sido exitoso.
Post condición	Un nuevo usuario existe ahora en la base de datos con su respectivo cargo.

No	17
Nombre/Identificador	CP_agregar2
Descripción	Agregar usuario combinando números y letras en el campo del documento.
Precondiciones	El administrador debe iniciar sesión para poder agregar a los demás usuarios.
Entradas	Número de documento del usuario que se quiere crear.
Pasos	1. Ingresar al sistema con los campos que piden, cargo, documento y contraseña. 2. Darle clic en la opción agregar usuarios. 3. Llenar todos los campos, y en el de documento escribir números y letras.
Resultados esperados	Mostrará un mensaje que diga "Ingrese solo números para el número de documento".
Post condición	No se puede agregar a un usuario porque en el campo del documento hay letras combinadas con números.

No	18
Nombre/Identificador	CP_agregar3
Descripción	Agregar usuario llenado solo los campos obligatorios.

Precondiciones	El administrador debe iniciar sesión para poder agregar a los demás usuarios.
Entradas	Datos en los campos que son obligatorios para crear un usuario.
Pasos	<ol style="list-style-type: none"> 1. Ingresar al sistema con los campos que piden, cargo, documento y contraseña. 2. Darle clic en la opción agregar usuarios. 3. Llenar los campos que son obligatorios por el sistema correctamente, dar clic en enviar y en ese momento se agregara un usuario. Los campos obligatorios son aquellos que contienen un asterisco dentro del campo respectivo.
Resultados esperados	Se muestra un mensaje que indica "Registro exitoso ".
Post condición	Se agregó un nuevo usuario.

No	19
Nombre/Identificador	CP_agregar4
Descripción	Agregar un usuario combinando números y letras en los campos de nombres y apellidos.
Precondiciones	El administrador debe iniciar sesión para poder agregar a los demás usuarios.
Entradas	Nombres y apellidos de los usuarios que se quieren crear.
Pasos	<ol style="list-style-type: none"> 1. Ingresar al sistema con los campos que piden, cargo, documento y contraseña. 2. Darle clic en la opción agregar usuarios. 3. Llenar todos los campos, y en los de nombres y apellidos escribir números y letras, y dar clic en enviar.
Resultados esperados	Se muestra un mensaje que indica: "Ingrese solo letras en los nombres y apellidos".
Post condición	No se puede agregar a un usuario porque en los campos de nombres y apellidos hay letras combinadas con números.

No	20
Nombre/Identificador	CP_agregar5
Descripción	Agregar usuario llenando solo los campos que no son obligatorios.
Precondiciones	El administrador debe iniciar sesión para poder agregar a los demás usuarios.
Entradas	Datos en los campos que no son obligatorios para crear un usuario.
Pasos	<ol style="list-style-type: none"> 1. Ingresar al sistema con los campos que piden, cargo, documento y contraseña. 2. Darle clic en la opción agregar usuarios.

	3. Llenar los campos que NO son obligatorios por el sistema y dar clic en enviar.
Resultados esperados	Se mostrarán mensajes que indican que se deben llenar los campos obligatorios.
Post condición	No se puede agregar a un usuario porque los campos que son obligatorios están vacíos.

No	21
Nombre/Identificador	CP_agregar6
Descripción	Agregar un usuario dejando los campos vacíos.
Precondiciones	El administrador debe iniciar sesión para poder agregar a los demás usuarios.
Entradas	Datos del usuario que se quiere crear.
Pasos	1. Ingresar al sistema con los campos que piden, cargo, documento y contraseña. 2. Darle clic en la opción agregar usuarios. 3. Dejar todos los campos vacíos, dar clic en enviar.
Resultados esperados	Se muestran varios mensajes que indican que se deben llenar, como mínimo, los campos que son obligatorios.
Post condición	No se puede agregar a un usuario porque los campos están vacíos.

2.6 Caso de pruebas - Recuperar contraseña

No	22
Nombre/Identificador	CP_recover_pass_001
Descripción	Recuperar contraseña enviando un correo a un e-mail existente.
Precondiciones	Tener un e-mail activo registrado en el sistema.
Entradas	E-mail del usuario.
Pasos	<ol style="list-style-type: none"> 1) Dirigir el cursor al botón de 'Iniciar sesión' y dar clic en este. 2) Dirigir el cursor al botón de '¿Olvidaste tu contraseña?' y dar clic en este. 3) Ingresar en el campo vacío el correo registrado a la cuenta de usuario a la que se quiere recuperar la contraseña. 4) Dirigir el cursor al botón de 'Enviar' y dar clic en este. 5) Dirigirse a la bandeja de entrada del correo ingresado y visualizar el correo de recuperar contraseña. 6) Copiar el enlace que se encuentra dentro del correo. 7) Pegar el enlace en la barra del buscador. 8) Rellenar los campos de contraseña nueva y de confirmación de contraseña. 9) Dirigir el cursor al botón de 'Guardar cambios' y dar clic en este.
Resultados esperados	Un recuadro de color verde aparecerá arriba de la cabecera de la página inicial para indicar que un correo ha sido enviado a la bandeja de entrada del usuario.
Post condición	El usuario usará su nueva contraseña para iniciar sesión y para cambiar su contraseña en caso de requerirlo.

No	23
Nombre/Identificador	CP_recover_pass_002
Descripción	Recuperar contraseña enviando un correo a un e-mail no existente en la base de datos.
Precondiciones	Iniciar sesión.
Entradas	E-mail del usuario
Pasos	<ol style="list-style-type: none"> 1) Dirigir el cursor al botón de 'Iniciar sesión' y dar clic en este. 2) Dirigir el cursor al botón de '¿Olvidaste tu contraseña?' y dar clic en este.

	3) Ingresar en el campo vacío un correo no existente en la base de datos del sistema. 4) Dirigir el cursor al botón de 'Enviar' y dar clic en este.
Resultados esperados	Un recuadro de color rojo aparecerá arriba de la cabecera de la página inicial, indicando que el correo electrónico no existe en la base de datos del sistema o que la estructura del mismo no es válida.
Post condición	El correo de recuperación de contraseña no será enviado al e-mail ingresado.

3. Pruebas manuales

3.1 Caso de prueba - Registrar un nuevo usuario

Código fuente

```
<div class="form-group">
  <!-- Se crea el input o campo en el que se digitará el primer nombre del usuario. El mismo cuenta con el tipo de dato respectivo,
  el nombre del campo y dos últimos atributos que permiten mostrar un tooltip, o caja con texto, que muestra información de interés al
  usuario: -->
  <input type="text" name="nombre1" class="form-control" id="nombre1" aria-describedby="emailHelp"
    placeholder="Primer nombre *" tool-tip-toggle="tooltip-required" data-original-title="Este campo es obligatorio">
  <!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo
  'validacion_registro.js' y que son referentes a las validaciones del campo de primer nombre: -->
  <div id="alerta"></div>
</div>
```

Caso 3.1.1: Datos de entrada: Registrar primer nombre del usuario. Resultado esperado(salida): el sistema guarda el primer nombre y lo manda al otro campo del registro.

```
<div class="form-group">
  <!-- Se crea el input o campo en el que se digitará el segundo nombre del usuario. El mismo cuenta con el tipo de dato respectivo y
  el nombre del campo: -->
  <input type="text" name="nombre2" class="form-control" id="nombre2" aria-describedby="emailHelp"
    placeholder="Segundo nombre">
  <!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo '
  validacion_registro.js' y que son referentes a las validaciones del campo de segundo nombre: -->
  <div id="alerta9"></div>
</div>
```

Caso 3.1.2: Datos de entrada: Registrar segundo nombre del usuario. Resultado esperado(salida): el sistema guarda el segundo nombre y lo manda al otro campo del registro.

```
<!-- Se indica, con una clase Bootstrap, que el elemento a añadir hace para de un formulario:-->
<div class="form-group">
  <!-- Se crea el input o campo en el que se digitará el primer apellido del usuario. El mismo cuenta con el tipo de dato respectivo, el nombre
  del campo y dos últimos atributos que permiten mostrar un tooltip, o caja con texto, que muestra información de interés al usuario: -->
  <input type="text" name="apellido1" class="form-control" id="apellido1" placeholder="Primer apellido *" tool-tip-toggle="tooltip-required"
    data-original-title="Este campo es obligatorio">
  <!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo '
  validacion_registro.js' y que son referentes a las validaciones del campo de primer apellido: -->
  <div id="alerta2"></div>
</div>
```

Caso 3.1.3: Datos de entrada: Registrar primer apellido del usuario. Resultado esperado(salida): el sistema guarda el primer apellido y lo manda al campo siguiente del registro.

```
<div class="form-group">
  <!-- Se crea el input o campo en el que se digitará el segundo apellido del usuario. El mismo cuenta con el tipo de dato respectivo y el
  nombre del campo: -->
  <input type="text" name="apellido2" class="form-control" id="apellido2" placeholder="Segundo apellido">
  <!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo
  'validacion_registro.js' y que son referentes a las validaciones del campo de segundo apellido: -->
  <div id="alerta8"></div>
</div>
```

Caso 3.1.4: Datos de entrada: Registrar segundo apellido del usuario. Resultado esperado(salida): el sistema guarda el segundo apellido y lo manda al campo siguiente del registro.

```
<div class="form-group">
  <!-- Se crea el input o campo en el que se digitará el número de documento del usuario. El mismo cuenta con el tipo de dato respectivo,
  el nombre del campo y dos últimos atributos que permiten mostrar un tooltip, o caja con texto, que muestra información de interés al usuario: -->
  <input type="text" name="documento" class="form-control" id="documento" aria-describedby="emailHelp" placeholder="Número de documento **">
  <!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo
  'validacion_registro.js' y que son referentes a las validaciones del campo de número de documento: -->
  <div id="alerta3"></div>
</div>
```

Caso 3.1.5: Datos de entrada: Registrar documento de identidad del usuario. Resultado esperado(salida): el sistema guarda el documento de identidad y lo manda al campo siguiente del registro.

```
<div class="form-group">
  <!-- Se crea el input o campo en el que se digitará el correo electrónico del usuario. El mismo cuenta con el tipo de dato respectivo,
  el nombre del campo y dos últimos atributos que permiten mostrar un tooltip, o caja con texto, que muestra información de interés al usuario: -->
  <input type="email" name="email" class="form-control" id="email" aria-describedby="emailHelp" placeholder="Correo electrónico **">
  <!-- Se añade un icono de Font Awesome que permite indicar al usuario, al hacer hover sobre el mismo, que el correo electrónico debe
  funcionar con el servidor de Gmail: -->
  <span class="mail icon">
    <i class="fa fa-envelope fa-lg" tool-tip-toggle="tooltip-email" data-original-title="Se requiere que el correo electrónico funcione
    on el servidor de Gmail, para así poder realizar la recuperación de la contraseña en caso de ser necesario. Por ejemplo, el correo
    'example1@gmail.com' cumple con lo requerido."></i>
  </span>
  <!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo
  'validacion_registro.js' y que son referentes a las validaciones del campo de correo electrónico: -->
  <div id="alerta4"></div>
</div>
```

Caso 3.1.6: Datos de entrada: Registrar el correo electrónico del usuario. Resultado esperado(salida): el sistema guarda el correo electrónico y lo manda al campo siguiente del registro.

```
<div class="form-group">
  <!-- Se crea un 'badge' o insignia de Bootstrap, la cual muestra un título dentro de un contenedor pequeño: -->
  <span id="perfil" class="badge font-weight-bold">Cargo </span>
  <!-- Se añade una etiqueta '<select>', la misma contiene las opciones para escoger el cargo que poseerá el nuevo usuario, así como una caja
  con texto informativa que es diferente a las demás ya mencionadas: -->
  <select name="cargo" class="form-control" tool-tip-toggle="tooltip-required" data-original-title="Escoja el cargo de acuerdo a la
  funcionalidad del usuario">
    <!-- Cargos disponibles en el formulario: -->
    <option>Instructor</option>
    <option>Personal administrativo</option>
    <option>Administrador</option>
  </select>
</div>
```

Caso 3.1.7: Datos de entrada: Registrar cargo al que pertenece el usuario. Resultado esperado(salida): el sistema guarda el cargo que seleccionó el usuario y lo manda al campo siguiente del registro.

```

<div class="form-group">
  <!-- Se crea el input o campo en el que se digitará la contraseña del usuario. El mismo cuenta con el tipo de dato respectivo, el nombre del
  campo y dos últimos atributos que permiten mostrar un tooltip, o caja con texto, que muestra información de interés al usuario: -->
  <input type="password" name="contraseña" class="form-control" id="contraseña"
    aria-describedby="emailHelp" placeholder="Contraseña *" tool-tip-toggle="tooltip-required" data-original-title="Este campo es obligatorio" />
  <!-- Se añade un icono de Font Awesome que, al hacer un 'hover' sobre el mismo, muestra una caja con texto que da información acerca de la
  contraseña al usuario -->
  <span class="info_icon">
    <i class="fa fa-info-circle fa-lg" tool-tip-toggle="tooltip-show" data-original-title="INFORMACIÓN: Por cuestiones de seguridad, la
    contraseña debe contener como mínimo: 8 caracteres, una mayúscula, una minúscula, un número y un caracter especial o raro. Por ejemplo,
    a contraseña 'Sena_1234' cumple con los parámetros requeridos."></i>
  </span>
  <!-- Se añade código JavaScript para los 'tooltips' o cajas con texto informativas: -->
  <script type="text/javascript">
    /* Se crea una función que posee diferentes eventos con código JQuery. Las mismas permiten, usando el atributo 'tool-tip-toggle', que una
    pequeña caja con texto pueda ser mostrada arriba, abajo, a la izquierda o a la derecha del elemento que posee el valor puesto dentro del
    atributo ya mencionado. Una vez el código HTML haya sido cargado correctamente, estas funciones podrán mostrarse haciendo 'hover' sobre el
    elemento que posee el atributo 'tool-tip-toggle': */
    $(document).ready(function(){
      // El valor 'tooltip-show' es llamado para mostrar una caja de texto con las recomendaciones para crear una contraseña segura:
      $('[tool-tip-toggle="tooltip-show"]').tooltip({
        // El posicionamiento del tooltip, en este caso, es hacia la derecha:
        placement : 'right'
      });
      // Aquí termina el código para este evento:
    });
    // Aquí inicia un nuevo evento para la función JQuery.
    // El valor 'tooltip-pass' es llamado para mostrar la caja de texto que indica que las contraseñas se pueden mostrar en el formulario
    //registro de un usuario:
    $('[tool-tip-toggle="tooltip-pass"]').tooltip({
      placement : 'right'
    });
    // El valor 'tooltip-required' indica con una caja de texto los campos que son obligatorios rellenar en el
  </script>

  //formulario de registro:
  $('[tool-tip-toggle="tooltip-required"]').tooltip({
    // El posicionamiento del tooltip, en este caso, es hacia la izquierda:
    placement : 'left'
  });
  // El valor 'tooltip-choose' indica en el formulario de registro que se debe escoger un cargo para el usuario
  // dependiendo de su funcionalidad en el sistema:
  $('[tool-tip-toggle="tooltip-choose"]').tooltip({
    placement : 'left'
  });
  // El valor 'tooltip-email' indica que el correo electrónico del usuario a registrar debe funcionar con el
  //servidor de Gmail:
  $('[tool-tip-toggle="tooltip-email"]').tooltip({
    placement : 'right'
  });
  // Aquí termina el código de la función JQuery.
</script>

  <!-- Se añade un icono de Font Awesome dentro de una clase para controlar sus estilos de manera más específica. Este icono indica, al hacer
  un 'hover' sobre él, que haciendo clic sobre el mismo se pueden visualizar las contraseñas ingresadas en los campos: -->
  <span class="show_icon">
    <i class="fa fa-eye fa-lg show" tool-tip-toggle="tooltip-pass" data-original-title="Mostrar contraseñas" id="show_password"></i>
  </span>
  <!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo
  'validacion_registro.js' y que son referentes a las validaciones del campo de contraseña: -->
  <div id="alerta5"></div>
</div>

```

Caso 3.1.8: Datos de entrada: Registrar la contraseña del usuario y poder visualizarla si así se desea. Resultado esperado(salida): el sistema guarda la contraseña del usuario y lo manda al campo siguiente del registro.

```


<!-- Se crea el input o campo en el que se digitará la confirmación de la contraseña del usuario. El mismo cuenta con el tipo de dato
respectivo, el nombre del campo y dos últimos atributos que permiten mostrar un tooltip, o caja con texto, que muestra información de
interés al usuario: -->
<input type="password" class="form-control" id="confirmar_contraseña" aria-describedby="emailHelp" placeholder="Confirmar contraseña "
  " tool-tip-toggle="tooltip-required" data-original-title="Este campo es obligatorio">
<!-- Se añade una etiqueta '<div>' con un identificador dentro, este permite mostrar las alertas que vienen desde el archivo
'validacion_registro.js' y que son referentes a las validaciones del campo de confirmar contraseña: -->
<div id="alerta6"></div>

<!-- Se usa código PHP para validar si el registro fue realizado con éxito o si, por el contrario, existen inconvenientes a la hora de
tratar de hacerlo: -->
<?php
// Se verifica si el valor 'new_user' está definido y se obtiene si el valor 'new_user' ubicado en el archivo 'ACTION_sign_up.php'
//devuelve true:
if (isset($_GET["new_user"]) && $_GET["new_user"] == 'true') {
    // En caso de ser así, se insertará código HTML en este archivo para realizar la aparición de una alerta tipo 'success' de bootstrap,
    // la cual aparecerá abajo del campo de confirmar contraseña. Esta alerta indica el registro exitoso de un nuevo usuario en el sistema
    echo "<div class= 'alert alert-success alert-dismissible fade show last_alerts' role='alert'><button type='button' class='close'
    data-dismiss='alert' aria-label='close'><span aria-hidden='true'>&times;</span></button>Registro de usuario exitos</div>";
}
// Se verifica si el valor 'double' está definido y se obtiene si el valor 'double' ubicado en el archivo 'ACTION_sign_up.php' devuelve
//true:
if (isset($_GET["double"]) && $_GET["double"] == 'true') {
    // En caso de ser así, la alerta, en este caso, es de tipo 'danger' e indica que tanto el número de documento y el correo electrónico
    //ya se encuentran registrados en el sistema, y, por lo tanto, no es posible hacer el registro:
    echo "<div class= 'alert alert-danger alert-dismissible fade show last_alerts' role='alert'><button type='button' class='close'
    data-dismiss='alert' aria-label='close'><span aria-hidden='true'>&times;</span></button>El documento y el correo electrónico
    ingresados ya se encuentran en la base de datos. Por favor corrija o ingrese un documento y un correo nuevos.</div>";
}
// Se verifica si el valor 'notavailable' está definido y se obtiene si el valor 'notavailable' ubicado en el archivo
// 'ACTION_sign_up.php' devuelve true:
if (isset($_GET["notavailable"]) && $_GET["notavailable"] == 'true') {
    // En caso de ser así, la alerta, en este caso, es de tipo 'danger' e indica que el número de documento ya se encuentra registrado
    //en la base de datos, y, por lo tanto, no es posible hacer el registro:
    echo "<div class= 'alert alert-danger alert-dismissible fade show last_alerts' role='alert'><button type='button' class='close'
    data-dismiss='alert' aria-label='close'><span aria-hidden='true'>&times;</span></button>El documento ingresado ya se encuentra en la
    base de datos. Por favor corrija o ingrese uno nuevo</div>";
}
// Si el anterior condicional devuelve false, entonces se verifica si el valor 'notemail' está definido y se obtiene si el valor 'notemail'
// ubicado en el archivo 'ACTION_sign_up.php' devuelve true:
elseif (isset($_GET["notemail"]) && $_GET["notemail"] == 'true') {
    // En caso de ser así, la alerta, en este caso, es de tipo 'danger' e indica que el correo electrónico ingresado ya está registrado
    //en la base de datos, y, por lo tanto, no es posible hacer el registro:
    echo "<div class= 'alert alert-danger alert-dismissible fade show last_alerts' role='alert'><button type='button' class='close'
    data-dismiss='alert' aria-label='close'><span aria-hidden='true'>&times;</span></button>El correo electrónico ingresado ya se encue
    en la base de datos. Por favor ingrese uno nuevo</div>";
}
?>
</div>


```

Caso 3.1.9: Datos de entrada: Confirmar la contraseña del usuario y poder visualizarla si así se desea. Resultado esperado(salida): el sistema guarda la contraseña del usuario y lo manda al botón “Enviar”.

```

<!-- Se añade, con la etiqueta '<input>', el botón que permitirá enviar el formulario una vez este se encuentre completado. Dicho botón
validará el formulario al darse click en él: -->
<input type="submit" value="Enviar" name="send" class="btn btn-md btn-block text-center"
  id="enter_button" onclick="validar_formulario()">

```

Caso 3.1.10: Datos de entrada: dar clic en el botón enviar. Resultado esperado(salida): el sistema guardara la información del usuario y se registra todo”.

4. Conclusiones.

Se realizaron las anteriores pruebas manuales para verificar el correcto funcionamiento de algunos módulos del sistema, para de esta manera poder entregar un producto totalmente funcional al usuario.