**INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**ASIGNATURA**

**RESIDENCIA**

**Segundo Reporte**

**TEMA DESARROLLADO**

**CONCLUSIÓN DE LAS RESPUESTAS**

**PARTICIPANTES:**

**NOMBRE Nº DE CONTROL**

Rafael Antonio Bolaños Cameras 16270740

Ricardo de Jesús González Morales 16270775

**NOMBRE DEL PROFESOR:**

**Aída Guillermina Cossío Martínez**

**e**

**LUGAR: FECHA:**

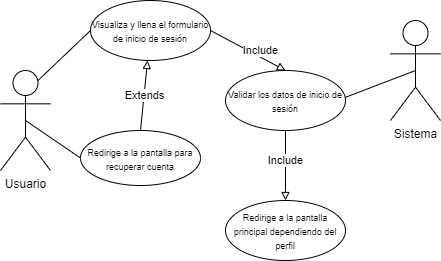
**17/11/2020**

**Tuxtla Gutiérrez, Chiapas**

## **1 Diseño del sistema**

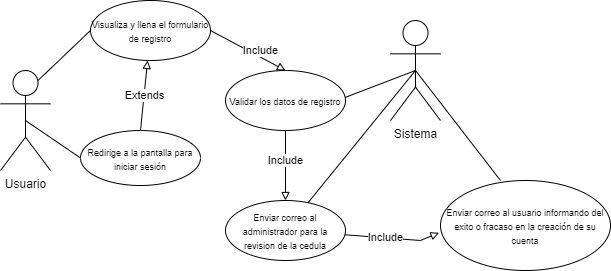
### **1.1 Casos de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| 1\_InicioSesion | | | |
| Nombre del caso de uso: | Inicio Sesión | | |
| Dependencias: | N/A | | |
| Actores: | Todos los usuarios | | |
| Tipo actor: | Primario | | |
| Precondición: | Contar con credenciales de accesos (usuario activo), y tener previo registro en el aplicativo. | | |
| Postcondición: | Tener acceso a la pantalla principal, dependiendo del perfil del usuario logueado (administrador o doctor) | | |
| Descripción: | El usuario podrá ingresar sus datos al formulario, validar los datos y dar acceso al menú principal dependiendo del perfil de usuario | | |
| Frecuencia: | Cada que el usuario desee acceder al aplicativo web PsychIA | | |
| Secuencia normal: | Paso | Acción | |
|  | 1 | Se visualiza la pantalla **01\_InicioSesion** | |
|  | 2 | El usuario debe ingresar los siguientes campos:   * Correo electrónico. (Type=email) * Contraseña. (Type= password) | |
|  | 3 | Clic sobre el botón Entrar o presionar la tecla Enter. | |
|  | 4 | El sistema valida los datos ingresados, que el usuario se encuentre registrado y con status activo | |
|  | 5 | Se deberá guardar los datos del usuario en una variable de sesión y posteriormente se redireccionará a la pantalla **3.1\_PantallaPrincipalDoctor** o **3.2\_PantallaPrincipalAdmin** (El menú que se deberá mostrar en esta pantalla dependerá del perfil del usuario logueado). | |
|  | 6 | Vista del menú que se deberá cargar según el perfil del usuario, de igual forma se deberá colocar la leyenda con el “Tipo de usuario (Perfil)” a la pantalla correspondiente (**3.1\_PantallaPrincipalDoctor** o **3.2\_PantallaPrincipalAdmin**)  **Notas**:   * El menú deberá ser fijo, al hacer uso del el scroll de la página no deberá de moverse * La opción salir se deberá visualizar en todos los perfiles, al final de las opciones * Cuando se presione sobre alguna de las opciones se deberá de subrayar de color verde como se ven en las pantallas correspondientes a cada opción del menú | |
| Excepciones: | Paso | Acción | |
|  | 2 y 3 | Validar datos con BootstrapValidator | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | * Campo **Correo**:   + Validar que no contenga espacios, solo letras, números y caracteres ASCII. Mensaje de error: “Campo correo tiene un formato no valido”.   + Validar que contenga datos. Mensaje de error: “Campo correo está vacío”. * Campo **Contraseña**:   + Validar que el campo contenga solo letras y números. Mensaje de error: “Campo contraseña solo acepta letras y números”.   + Validar que contenga un máximo de 10 caracteres. Mensaje de error: “Campo contraseña solo acepta máximo 10 caracteres”.   + Validar que contenga datos. Mensaje de error: “Campo contraseña está vacío”.   **\*Campos requeridos** |
|  |  | 2 | En caso de generarse cualquiera de los errores antes mencionados, se deberá marcar de color rojo el contorno de el/los campos incorrectos y mostrar los mensajes en forma de lista. Cuando el usuario ingrese los caracteres correctos, se deberá eliminar el/los mensaje de error en la lista y pintar de color verde el contorno de el/los campos correctos.  **Nota:** Las validaciones se deberán realizar en tiempo real. |
|  | 2 | Clic sobre la etiqueta ¿Olvidaste tu contraseña? | |
|  |  | Número de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Redireccionar a la pantalla **2\_RecuperarCuenta** |
|  | 4 | Validar datos con python | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Se validará cada campo de los formularios que sean enviados a la base de datos. |
|  |  | 2 | Se deberá validar que no pueda realizarse la inyección SQL mediante la seguridad CSRF. |
|  | 4 | Si los datos de acceso no concuerdan con alguno en la base de datos. | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Notificar al usuario mediante un Modal con el mensaje “No existe registro con esos datos. Verifícalos” |
|  |  | 2 | Al presionar Aceptar, se deberá cerrar el **Modal,** y mantener al usuario en el formulario. |
|  | 4 | Si la cuenta o el usuario no se encuentra activo | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Notificar al usuario mediante un **Modal** con el mensaje “La cuenta con la que desea entrar no está activa, favor de contactar con un administrador”. |
|  |  | 2 | Al presionar Aceptar, se deberá cerrar el **Modal,** y mantener al usuario en el formulario. |
|  | 6 | Clic sobre el logotipo PysquIA | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Deberá redireccionar a la pantalla **3.1\_PantallaPrincipalDoctor o 3.2\_PantallaPrincipalAdmin** dependiendo del usuario |
|  | 6 | Clic sobre alguna de las opciones del menú | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Se deberá redirigir a la pantalla correspondiente. |
|  | 4, 6 | Error de comunicación al aplicativo | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Notificar al usuario mediante un **Modal,** “Se produjo problema de comunicación. Vuelva Intentarlo más tarde”. |
| Comentarios | * Se deberá validar que el aplicativo sea compatible con los principales navegadores como Opera, Google Chrome, Internet Explorer, Mozilla, etc. * Crear un hash con el campo contraseña mediante SHA-1 y posteriormente comprara con la almacenada en la base de datos. * Cuando el usuario presione la tecla “Enter” sobre algún campo se deberá aplicar las validaciones antes mencionadas. * Todos los botones del formulario deberán usar la clase btn de Bootstrap, esto para generar el efecto hover al pasar el puntero sobre el botón. * Todos los campos que se envíen a la base de datos, deberán utilizar la clase validacionSqlInjection (se deberá validar que no contenga los símbolos: Ã±, Ã,’, Â, \\\\, \\{,;,Âµ,â€,\001,\001,}), mediante el método **reeplaceCaracter** de la clase, pasándole como parámetro la cadena a limpiar. * Cuando el usuario de clic sobre algún botón o acción del formulario, mostrar en pantalla un **loader**, para indicar al usuario que su petición se está procesando. * El aplicativo deberá ser “Responsive”, y tendrá que ser adaptable para tablets, laptop y de escritorio. | | |



**Ilustración 1 - Diagrama para inicio de sesión**

|  |  |  |  |
| --- | --- | --- | --- |
| 2\_RegistroDeUsuarios | | | |
| Nombre del caso de uso: | Registro de Usuarios | | |
| Dependencias: | N/A | | |
| Actores: | Todos los usuarios | | |
| Tipo actor: | Primario | | |
| Precondición: | Contar con una cedula profesional activa con el perfil de psicólogo o psiquiatra. | | |
| Postcondición: | Redireccionar a la pantalla de inicio de sesión | | |
| Descripción: | El usuario podrá ingresar sus datos al formulario, validar los datos y registrar su cuenta con éxito. | | |
| Frecuencia: | Cada que un nuevo doctor desee tener acceso al aplicativo web. | | |
| Secuencia normal: | Paso | Acción | |
|  | 1 | Se visualiza la pantalla **02\_Registro** | |
|  | 2 | El usuario debe ingresar los siguientes campos:   * Nombres. (Type=string) * Apellidos. (Type=string) * Cedula. (Type=string) * Correo. (Type=email) * Contraseña. (Type= password) * Repetir Contraseña. (Type=password) | |
|  | 3 | Clic sobre el botón Entrar o presionar la tecla Registrar Cuenta. | |
|  | 4 | El sistema valida los datos ingresados, que todos los campos estén llenados con datos correctos validados con bootstrapValidator. | |
|  | 5 | Se deberá enviar un correo al administrador para que este verifique que la cedula sea existente y procederá a aceptar la cuenta para dar acceso al nuevo doctor. | |
|  | 6 | Se envía un correo al usuario una vez el administrador verifique que todo esta en orden para que este pueda ser aceptado en el aplicativo. | |
| Excepciones: | Paso | Acción | |
|  | 2 y 3 | Validar datos con BootstrapValidator | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | * Campo **Nombres:**   + Validar que no contenga espacios, solo letras. Mensaje de error: “Campo nombres tiene un formato no valido”.   + Validar que contenga datos. Mensaje de error: “Campo correo está vacío”. * Campo **Apellidos:**   + Validar que no contenga espacios, solo letras. Mensaje de error: “Campo nombres tiene un formato no valido”.   + Validar que contenga datos. Mensaje de error: “Campo correo está vacío”. * Campo **Cedula:**   + Validar que no contenga espacios, solo letras y caracteres ASCII. Mensaje de error: “Campo nombres tiene un formato no valido”.   + Validar que contenga datos. Mensaje de error: “Campo correo está vacío”. * Campo **Correo**:   + Validar que no contenga espacios, solo letras, números y caracteres ASCII. Mensaje de error: “Campo correo tiene un formato no valido”.   + Validar que contenga datos. Mensaje de error: “Campo correo está vacío”. * Campo **Contraseña**:   + Validar que el campo contenga solo letras y números. Mensaje de error: “Campo contraseña solo acepta letras y números”.   + Validar que contenga un máximo de 10 caracteres. Mensaje de error: “Campo contraseña solo acepta máximo 10 caracteres”.   + Validar que contenga datos. Mensaje de error: “Campo contraseña está vacío”.   **\*Campos requeridos** |
|  |  | 2 | En caso de generarse cualquiera de los errores antes mencionados, se deberá marcar de color rojo el contorno de el/los campos incorrectos y mostrar los mensajes en forma de lista. Cuando el usuario ingrese los caracteres correctos, se deberá eliminar el/los mensaje de error en la lista y pintar de color verde el contorno de el/los campos correctos.  **Nota:** Las validaciones se deberán realizar en tiempo real. |
|  | 3 | Clic sobre la etiqueta Ya tienes una cuenta? Iniciar Sesión | |
|  |  | Número de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Redireccionar a la pantalla 01**\_InicioSesion** |
|  | 4 | Validar datos con python | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Se validará cada campo de los formularios que sean enviados a la base de datos. |
|  |  | 2 | Se deberá validar que no pueda realizarse la inyección SQL mediante la seguridad CSRF. |
|  | 5 | La cédula no existe o contiene algún error | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Notificar al usuario mediante un correo indicando que la cédula otorgada tiene errores y que proceda con el registro nuevamente |
|  |  | 2 | Se elimina el pre-registro de la base de datos. |
|  | 4 | Si la cuenta o el usuario no se encuentra activo | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Notificar al usuario mediante un **Modal** con el mensaje “La cuenta con la que desea entrar no está activa, favor de contactar con un administrador”. |
|  |  | 2 | Al presionar Aceptar, se deberá cerrar el **Modal,** y mantener al usuario en el formulario. |
|  | 3 | Clic sobre el logotipo PysquIA | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Deberá redireccionar a la pantalla de inicio general. |
|  | 4, 6 | Error de comunicación al aplicativo | |
|  |  | Numero de secuencia para manejar la excepción. | Descripción. |
|  |  | 1 | Notificar al usuario mediante un **Modal,** “Se produjo problema de comunicación. Vuelva Intentarlo más tarde”. |
| Comentarios | * Se deberá validar que el aplicativo sea compatible con los principales navegadores como Opera, Google Chrome, Internet Explorer, Mozilla, etc. * Crear un hash con el campo contraseña mediante SHA-1 y posteriormente comprara con la almacenada en la base de datos. * Cuando el usuario presione la tecla “Enter” sobre algún campo se deberá aplicar las validaciones antes mencionadas. * Todos los botones del formulario deberán usar la clase btn de Bootstrap, esto para generar el efecto hover al pasar el puntero sobre el botón. * Todos los campos que se envíen a la base de datos, deberán utilizar la clase validacionSqlInjection (se deberá validar que no contenga los símbolos: Ã±, Ã,’, Â, \\\\, \\{,;,Âµ,â€,\001,\001,}), mediante el método **reeplaceCaracter** de la clase, pasándole como parámetro la cadena a limpiar. * Cuando el usuario de clic sobre algún botón o acción del formulario, mostrar en pantalla un **loader**, para indicar al usuario que su petición se está procesando. * El aplicativo deberá ser “Responsive”, y tendrá que ser adaptable para tablets, laptop y de escritorio. | | |



**Ilustración 2 - Diagrama Registro de cuenta**

### **1.2 Vistas del sistema**

#### **Página de Inicio General**

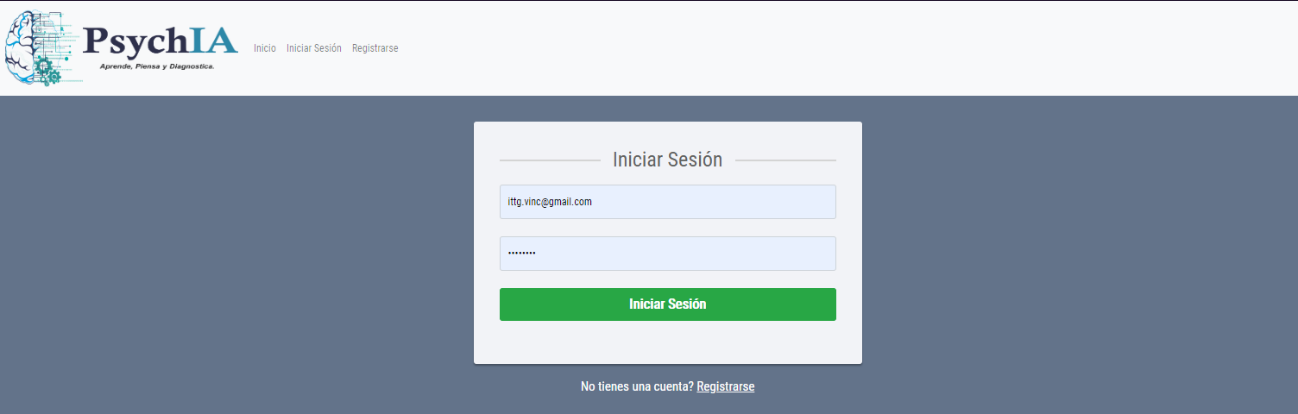


Esta pantalla será accesible para cualquier tipo de usuario, se tendrá una descripción breve del propósito del aplicativo, quienes somos como creadores del sistema, las redes sociales y el botón para iniciar sesión.

Los botones de las redes sociales a las páginas creadas para hacer publicidad mediante las mismas redes.

El botón *Mi cuenta* direccionara a las pantallas de inicio de sesión.

#### **Página de Inicio de Sesión**

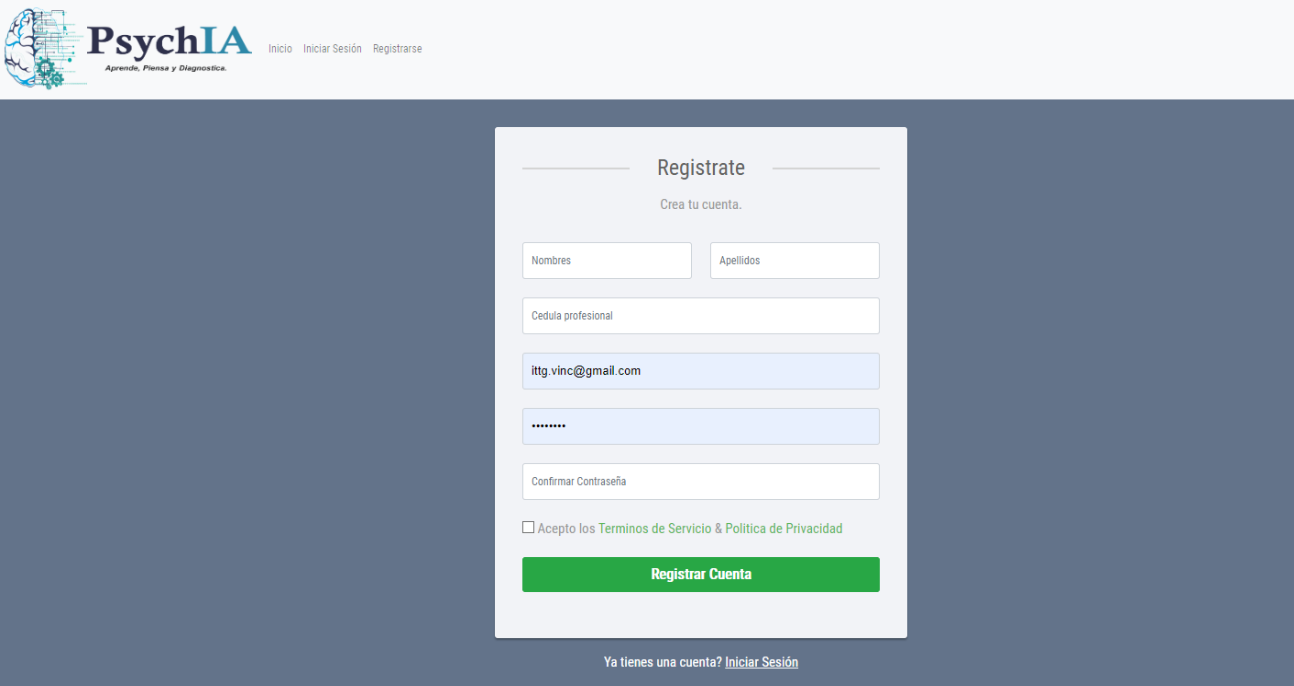
. 

Este template será el encargado de mostrar el formulario de inicio de sesión de los usuarios. Esto únicamente tendrá los inputs que soliciten el correo y la contraseña del usuario.

Estos campos estarán relacionados a la validación de Bootstrap para validar en tiempo real en el cliente que lo que se esté escribiendo cumpla con los requerido en los casos de uso. Una vez se de click al botón *Iniciar Sesión* se procederá con la validación a nivel vista revisando que el usuario exista en la base de datos y sea la misma contraseña que fue proporcionada.

Si todo es correcto pasará a la pantalla principal dependiendo del rol que tenga el usuario.

#### **Página de Registro**



El template del registro mostrará el formulario con los campos solicitados en el caso de uso correspondiente. Estos campos serán validados en tiempo real del lado del cliente al igual que sucede con la pantalla de *Inicio de Sesión* usando bootstrapValidator.

Hay dos botones extras, uno para la lectura de los términos y condiciones del uso del aplicativo y otro mostrando las políticas de privacidad que estaremos manejando. Esto únicamente abrirá modales de Bootstrap dependiendo del enlace que se de clic. Al dar clic a *Registrar Cuenta* se procederá a la validación a nivel *vista* checando que el usuario creado no exista en la base de datos.

Este registro será mandado al administrador para validar la cédula de manera individual para decidir si la persona puede o no usar el aplicativo.