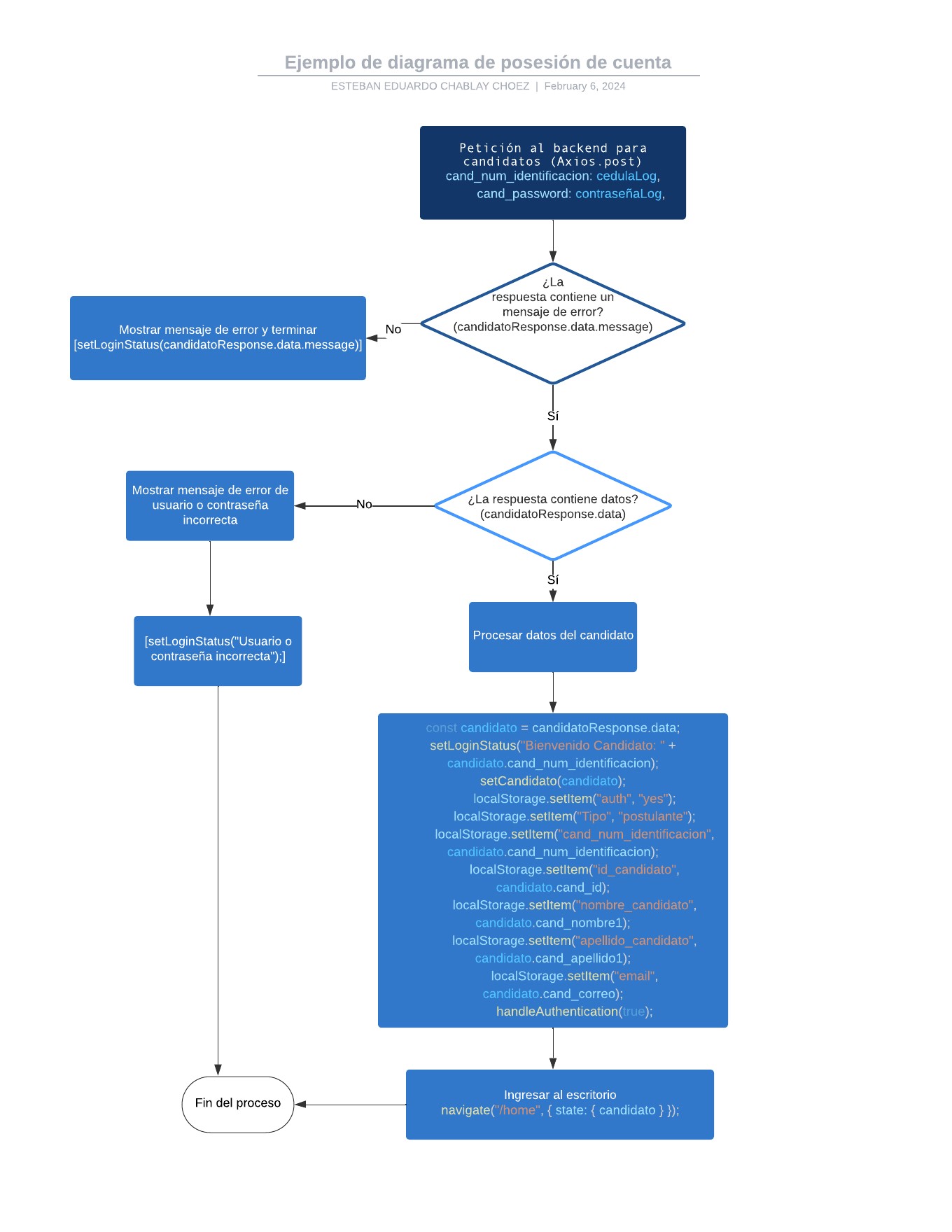
|  |
| --- |
|  |
| Prueba de Caja Blanca |
| ***“Sistema de Postulación - ESPE”*** |
| Versión 1.1 |
|  |
| **Integrantes:**  Esteban Chablay Adrian Mosquera Fernando Paredes  Santiago Risueño  **Fecha 06/02/2024** |

**BUSCAR PROBLEMAS**

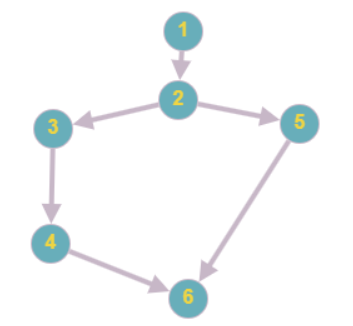
**CÓDIGO FUENTE**

|  |
| --- |
| // Lógica de Correos  useEffect(() => emailjs.init("eT9gXHINase6VFYsd"), []); // Replace "user\_your\_emailjs\_user\_id" with your actual EmailJS user ID  const handleSubmit = async (e) => {  console.log(nameRef.current.value, emailRef.current.value);  const serviceId = "service\_xjgs5nf"; // Replace with your actual service ID  const templateId = "template\_joyhl86"; // Replace with your actual template ID  try {  setLoading(true);  const newConfirmationCode = generateConfirmationCode();  setConfirmationCode(newConfirmationCode);  const response = await emailjs.send(  serviceId,  templateId,  {  to\_name: emailRef.current.value,  recipient: nameRef.current.value,  cedula: cedula,  titulo: tituloRef.current.value,  CodigoConfirmacion: newConfirmationCode,  }  );  if (response.status === 200) {  //alert("Email successfully sent, check your inbox.");  } else {  console.log("Error sending email:", response.text);  //alert("Failed to send the email. Please try again later.");  }  } catch (error) {  console.log("Error sending email:", error);  //alert("Failed to send the email. Please try again later.");  } finally {  setLoading(false);  }  };  const handleAcceptClick = () => {  handleCodigoClick(); // Llama a la función Codigo  handleSubmit(); // Llama a la función handleSubmit  setShowConfirmModal(false)  };  const handleAcceptCodigoClick = () => {  if (userConfirmationCode === confirmationCode) {  register();  } else {  alert("Código de confirmación incorrecto. Por favor, verifique.");  }  }; |

**Diagrama de Flujo**



**GRAFO**

****

**NODOS:**

1. Inicio
2. Petición al backend para registrar candidato (Axios.post)
3. ¿La petición es exitosa?
4. Actualizar el almacenamiento local y redirigir a la página de inicio
5. Mostrar error en la consola
6. Fin

**RUTAS**

**R1:** 1 -> 2 -> 3 -> 4 -> 6

**R2:** 1 -> 2 -> 5 -> 6

**COMPLEJIDAD CICLOMÁTICA**

La complejidad ciclomática se calcula como E - N + 2P, donde:

* E es el número de aristas
* N es el número de nodos
* P es el número de componentes conectados

Para este grafo:

* E = 5 (número de aristas)
* N = 6 (número de nodos)
* P = 1 (un solo componente conectado)

Por lo tanto, la complejidad ciclomática es:

C = 5 - 6 + 2\*1 = 1