|  |
| --- |
|  |
| Prueba de Caja Blanca |
| ***“Reinas ESPE”*** |
|  |
|  |
| **Integrantes:**  Luca De Veintemilla  Dylan Hernández  Juan Reyes  Kevin Vargas  **Fecha 2024-15-02** |

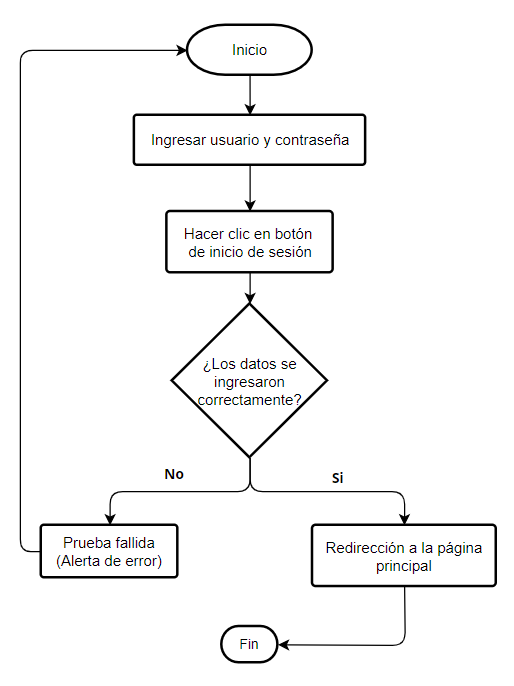
**BUSCAR PROBLEMAS**

**REQ001**

**CÓDIGO FUENTE**

|  |
| --- |
| <button className="loginBut" value="Iniciar Sesión" onClick={handleSubmit}>                <p>Iniciar Sesión</p>              </button>              <p className="iniciar-sesion"> © - Derechos Reservados</p>              <p className="iniciar-sesion"> Dpto. de Ciencias de la Computación</p>              <Popup open={modalIsOpen} onClose={handleModalClose}>                <div className="modal">                  <h2 className="modal-title">Error de Ingreso</h2>                  <p className="modal-p">{err}</p>                  <div className="botones-modal">                    <button onClick={() => { handleModalClose();}} className="btn-confirmar">                      Aceptar                    </button> |

**DIAGRAMA DE FLUJO**

****

**GRAFO**

**Imagen que contiene objeto

Descripción generada automáticamente**

**RUTAS**

**R1:** 1, 2, 3, 4, 6

**R2:** 1, 2, 3, 4, 5, 7

**COMPLEJIDAD CICLOMÁTICA**

Se puede calcular de las siguientes formas:

* V(G) = número de nodos predicados(decisiones)+1

V(G)= 1 + 1 = 2

* V(G) = A – N + 2

V(G)= 7 – 7 + 2 = 2

**DONDE:**

**P:** Número de nodos predicado

**A:** Número de aristas

**N:** Número de nodos

**REQ002**

**CÓDIGO FUENTE**

|  |
| --- |
| <div key={reinaIndex} className={`slide-container ${index === reinaIndex ? 'active' : ''}`}>                          <div className="slide">                              <div className="content">                                  <h3 className="circuloReferencia">{reina.CANDIDATA\_ID}</h3>                                  <h3 className="carrusel-heading">{reina.CAND\_NOMBRE1} {reina.CAND\_APELLIDOPATERNO}</h3>                                  <b><p className="carrusel-paragraph-dep">{reina.DEPARTMENTO\_NOMBRE}</p></b>                                  <p className="carrusel-paragraph"><b>Fecha de Nacimiento:</b> {fechaFormato(reina.CAND\_FECHANACIMIENTO.split('T')[0])} </p>                                  <p className="carrusel-paragraph"><b>Estatura:</b> {reina.CAND\_ESTATURA}</p>                                  <p className="carrusel-paragraph"><b>Idiomas:</b> {reina.CAND\_IDIOMAS}</p>                                  <p className="carrusel-paragraph"><b>Color de ojos:</b> {reina.CAND\_COLOROJOS}</p>                                  <p className="carrusel-paragraph"><b>Color de cabello:</b> {reina.CAND\_COLORCABELLO}</p><p className='carrusel-paragraph'> </p>                                  <p className="carrusel-paragraph"><b>Hobbies:</b></p><div className="carrusel-paragraph" dangerouslySetInnerHTML={{ \_\_html: reina.CAND\_HOBBIES }}></div>                              </div>                              <img src={imagenArray[reinaIndex]} alt="" />                          </div>                          {listaEvento.map((evento, eventoIndex) => {                              if (evento.EVENTO\_ID === 1 && evento.EVENTO\_ESTADO === "si") {                                  return (                                      <div class="boton">                                          <Link to="/TrajeTipico">                                              <button className='btn'>Iniciar Votación</button>                                          </Link>                                      </div>                                  );                              } else {                                  return null; // O puedes devolver un componente vacío (<></>)                              }                          })}                      </div> |

**DIAGRAMA DE FLUJO**

Diagrama

Descripción generada automáticamente

**GRAFO**

Imagen que contiene Diagrama

Descripción generada automáticamente

**RUTAS**

**R1:** 1, 2, 3, 4

**COMPLEJIDAD CICLOMÁTICA**

Se puede calcular de las siguientes formas:

* V(G) = número de nodos predicados(decisiones)+1

V(G)= 0 + 1 = 1

* V(G) = A – N + 2

V(G)= 4 – 4 + 2 = 2

**DONDE:**

**P:** Número de nodos predicado

**A:** Número de aristas

**N:** Número de nodos

**REQ003**

**CÓDIGO FUENTE**

|  |
| --- |
| const handleSubmit = async (e) => {      e.preventDefault();      try {        await login(inputs);        //await axios.put(`${API\_BASE\_URL}/user/${inputs.username}`);        if (inputs.username === "Veedor") {          navigate("/TablaNotario");        } else if (inputs.username === "admin" || inputs.username === "dylan" || inputs.username === "mathias") {          navigate("/PanelAdmin");        }        else {          navigate("/Carrusel");        }      } catch (err) {        setModalIsOpen(true);        setError(err.response.data);        console.log(err.response.data);      }    }; |

**DIAGRAMA DE FLUJO**

**Diagrama

Descripción generada automáticamente**

**GRAFO**

Forma

Descripción generada automáticamente con confianza media

**RUTAS**

**R1:** 1, 2, 3, 4, 6, 7

**R2:** 1, 2, 3, 4, 5, 7

**COMPLEJIDAD CICLOMÁTICA**

Se puede calcular de las siguientes formas:

* V(G) = número de nodos predicados(decisiones)+1

V(G)= 0 + 1 = 1

* V(G) = A – N + 2

V(G)= 7 – 7 + 2 = 2

**DONDE:**

**P:** Número de nodos predicado

**A:** Número de aristas

**N:** Número de nodos

**REQ004**

**CÓDIGO FUENTE**

|  |
| --- |
| <React.Fragment>  {listaReinas.map((item1, index) => {  return (  <div className="matrix">  <div>  <br></br>  <h1>{item1.CAND\_NOMBRE1} {item1.CAND\_APELLIDOPATERNO}</h1>  </div>  <div>  <section class="wrapper">  <main class="row title">  <ul>  <li>Juez</li>  <li>Traje Típico</li>  <li>Traje Gala</li>  <li>Barra</li>  </ul>  </main>  <section class="row-fadeIn-wrapper">  <article class="row fadeIn nfl">  {listaJueces.map((item2, index) => {      return (          <ul>              <li>{item2.id} - {item2.name}</li>              {votaciones.map((item3, index) => {                  if (item3.CANDIDATA\_ID === item1.CANDIDATA\_ID && item3.USUARIO\_ID === item2.id) {                      return (                          <li className={item3.VOT\_ESTADO === "no" ? "celda-roja" : "celda-verde"}>                              {item3.VOT\_ESTADO === "no" ? "Pendiente" : "Votado"}                          </li>                      );                  }              })}          </ul>      );  })}  </article>  </section>  </section>  </div>  </div>  );  }  )}  </React.Fragment> |

**DIAGRAMA DE FLUJO**

Diagrama

Descripción generada automáticamente

**GRAFO**

Imagen que contiene Diagrama

Descripción generada automáticamente

**RUTAS**

**R1:** 1, 2, 3, 4

**COMPLEJIDAD CICLOMÁTICA**

Se puede calcular de las siguientes formas:

* V(G) = número de nodos predicados(decisiones)+1

V(G)= 0 + 1 = 1

* V(G) = A – N + 2

V(G)= 4 – 4 + 2 = 2

**DONDE:**

**P:** Número de nodos predicado

**A:** Número de aristas

**N:** Número de nodos