

## ISWD652 CALIDAD DE SOFTWARE

**NOMBRE ESTUDIANTES:** Cabezas Erick  
Hernández Christian  
Jiménez Alejandro  
Rosillo Bryan  
Segovia Jorge

**FECHA:** 12-12-2023

**TEMA:** Traductor Asanas

---

### APRENDIZAJE OBTENIDO

Previo al desarrollo de los casos de prueba del programa realizado por el equipo DarkChococrispis Inc., se ha obtenido un aprendizaje sobre la mejora de la calidad de software, poniendo a prueba cada uno de los casos que se pueden dar durante la ejecución del programa, además se ha aprendido la sintaxis adecuada del desarrollo de casos de prueba, como lo es la tabla presentada, los resultados esperados y los obtenidos, siendo estos sustentos de las pruebas realizadas.

### DESARROLLO

#### CASO DE PRUEBA 1

ID	Descripción	Supuestos y Condiciones previas
CP01	Se proporciona al método un archivo de texto válido con datos en un formato correcto	El programa debe estar en ejecución y el archivo dentro de la carpeta raíz del proyecto.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El Arraylist correspondiente debe encontrarse con los datos que se encontraban dentro del archivo de texto proporcionado.

## ISWD652 CALIDAD DE SOFTWARE

```

this.posturas = (ArrayList@2491) size = 20
> {0} = (PosturaAsana@2591) "Traducción de la POSTURA: Parivrtta Trikonasana" Ingles: Revolved Triangle Pose Español: Postura del Triángulo Invertido Traducción del MORFEMA: Pariv... View
> {1} = (PosturaAsana@2613) "Traducción de la POSTURA: Tadasana" Ingles: Mountain Pose Español: Postura de la Montaña Traducción del MORFEMA: Tada Ingles = mountain... View
> {2} = (PosturaAsana@2623) "Traducción de la POSTURA: Adho Mukha Svanasana" Ingles: Downward-Facing Dog Español: Perro mirando hacia abajo Traducción del MORFEMA: adho... View
> {3} = (PosturaAsana@2633) "Traducción de la POSTURA: Virabhadrasana I" Ingles: Warrior I Español: Guerrero Traducción del MORFEMA: Virabhadra Ingles = warrior... Es... View
> {4} = (PosturaAsana@2642) "Traducción de la POSTURA: Virabhadrasana II" Ingles: Warrior II Español: Guerrero Traducción del MORFEMA: Virabhadra Ingles = warrior... E... View
> {5} = (PosturaAsana@2655) "Traducción de la POSTURA: Vrikshasana" Ingles: Tree Pose Español: Postura del Árbol Traducción del MORFEMA: Vriksha Ingles = tree... Es... View
> {6} = (PosturaAsana@2664) "Traducción de la POSTURA: Balasana" Ingles: Child's Pose Español: Postura del Niño Traducción del MORFEMA: Bala Ingles = kid... Español = n... View
> {7} = (PosturaAsana@2678) "Traducción de la POSTURA: Ardha Bhujangasana" Ingles: Half Cobra Pose Español: Postura de la Cobra a la mitad Traducción del MORFEMA: ardha... View
> {8} = (PosturaAsana@2686) "Traducción de la POSTURA: Paschimottanasana" Ingles: Seated Forward Bend Español: Flexión hacia adelante sentado Traducción del MORFEMA: Paschim... View
> {9} = (PosturaAsana@2696) "Traducción de la POSTURA: Setu Bandhasana" Ingles: Bridge Pose Español: Postura del Puente Traducción del MORFEMA: Setu Ingles = bridge... View
> {10} = (PosturaAsana@2705) "Traducción de la POSTURA: Shavasana" Ingles: Corpse Pose Español: Postura del Cadáver Traducción del MORFEMA: Shava Ingles = corpse... View
> {11} = (PosturaAsana@2715) "Traducción de la POSTURA: Trikonasana" Ingles: Triangle Pose Español: Postura del Triángulo Traducción del MORFEMA: Trikona Ingles = triangle... View
> {12} = (PosturaAsana@2725) "Traducción de la POSTURA: Kumbhakasana" Ingles: Plank Pose Español: Postura de la Plancha Traducción del MORFEMA: Kumbhaka Ingles = brea... View
> {13} = (PosturaAsana@2735) "Traducción de la POSTURA: Matsyasana" Ingles: Fish Pose Español: Postura del Pez Traducción del MORFEMA: Matsya Ingles = fish... Español... View
> {14} = (PosturaAsana@2745) "Traducción de la POSTURA: Utkatasana" Ingles: Chair Pose Español: Postura de la Silla Traducción del MORFEMA: Utkata Ingles = intense... Es... View
> {15} = (PosturaAsana@2755) "Traducción de la POSTURA: Malasana" Ingles: Garland Pose Español: Postura de la Guirnalda Traducción del MORFEMA: Mala Ingles = wreath... View
> {16} = (PosturaAsana@2764) "Traducción de la POSTURA: Eka Pada Rajakapotasana" Ingles: Pigeon Pose Español: Postura de la Paloma Traducción del MORFEMA: Eka Ingles = ... View
> {17} = (PosturaAsana@2772) "Traducción de la POSTURA: Urdhva Hastasana" Ingles: Upward Salute Español: Saludo hacia arriba Traducción del MORFEMA: urdhva Ingles = up... View
> {18} = (PosturaAsana@2781) "Traducción de la POSTURA: Parivrtta Trikonasana" Ingles: Revolved Triangle Pose Español: Postura del Triángulo Invertido Traducción del MORFEMA: Pariv... View
> {19} = (PosturaAsana@2790) "Traducción de la POSTURA: Parivrtta Janu Sirsasana" Ingles: Revolved Head To Knee Pose Español: Postura de la Cabeza a la Rodilla Invertida Traducción ... View

```

**Resultado real y condiciones:** El resultado real de la ejecución del programa se puede ver en la imagen anterior, en donde se muestra que el arraylist “this.posturas” se encuentra en efecto con todos los datos encontrados en el txt sin encontrarse error alguno al ejecutar.

### CASO DE PRUEBA 2

ID	Descripción	Supuestos y Condiciones previas
CP02	Se proporciona al método un texto válido pero el mismo se encuentra vacío, entonces, la clase no debe agregar ningún tipo de postura.	El programa debe estar en ejecución y el archivo dentro de la carpeta raíz del proyecto.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El arraylist de posturas debe encontrarse vacío.

```

{...} this = {Diccionario@2489}
posturas = {ArrayList@2491} size = 0

```

**Resultado real y condiciones:** El resultado real muestra en el Debugger como al proporcionar un archivo de texto vacío, simplemente el arraylist se mantiene con longitud 0, es decir, vacío, en lugar de mostrarse cualquier tipo de error.

### CASO DE PRUEBA 3

ID	Descripción	Supuestos y Condiciones previas
----	-------------	---------------------------------

## ISWD652 CALIDAD DE SOFTWARE

CP03	Se proporciona al método un archivo de texto inexistente, entonces, se debería manejar la excepción IOException y la clase no debería verse modificada.	El programa debe estar en ejecución y el archivo no debe existir en la carpeta raíz del proyecto.
<b>Fecha</b>	<b>Prueba Ejecutada Por</b>	<b>Resultado</b>
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debería mostrar en consola un mensaje “Archivo no encontrado” y el arraylist debería seguir vacío.



```

Debug Main x
Debugger Console
C:\Users\USER\.jdk\openjdk-20.0.2\bin\java.exe -agentlib:jdwp=transport=dt
Connected to the target VM, address: '127.0.0.1:52408', transport: 'socket'
Archivo no encontrado

```

```

{...} this = {Diccionario@2489}
posturas = {ArrayList@2491} size = 0

```

**Resultado real y condiciones:** El resultado real fue la impresión del mensaje correspondiente al manejar la excepción, además de no verse modificado el arraylist que se manipula de existir el archivo de texto.

### CASO DE PRUEBA 4

ID	Descripción	Supuestos y Condiciones previas
CP04	Se proporciona al método un archivo de texto el cual consta con una mala estructura para que se extraigan los datos a la clase de manera respectiva, entonces, la clase debe quedar vacía y la excepción controlada.	El programa debe estar en ejecución y el archivo debe existir en la carpeta raíz del proyecto con una estructura inadecuada intencionalmente.
<b>Fecha</b>	<b>Prueba Ejecutada Por</b>	<b>Resultado</b>
10-12-2023	Alejandro Jiménez	Fallido

**Resultado esperado:** El arraylist no debe ingresar los datos proporcionados en dicho archivo de texto, puesto que, la estructura está mal proporcionada, es decir, debería permanecer vacío.

## ISWD652 CALIDAD DE SOFTWARE

```

{...} this = {Diccionario@2489}
  {...} posturas = {ArrayList@2491} size = 1
    {...} 0 = {PosturaAsana@2587} Method threw 'java.lang.NullPointerException' exception. Cannot evaluate ec.edu.epn.PosturaAsana.toString()
      sanscrito = null
      ingles = null
      español = null
      palabrasBase = null
  
```

**Resultado real y condiciones:** El resultado real fue el añadido de los datos dentro del arraylist, la clase Gson, al no encontrar la estructura adecuada, lo que hizo fue añadir como “null” los campos correspondientes, por ende, al tratar de transformar la clase a String con el método “toString” existe el error “NullPointerException”.

### CASO DE PRUEBA 5

ID	Descripción	Supuestos y Condiciones previas
CP05	Se proporciona al método un archivo de texto que contiene una línea, entonces, la misma debe ser leída y agregada a la clase respectivamente. Debe suceder lo mismo si el archivo contiene n líneas.	El programa debe estar en ejecución y el archivo debe existir en la carpeta raíz del proyecto.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El método debe leer línea por línea al archivo de texto y almacenarla en el arraylist sin errores.

```

public void obtenerPosturas() {
    try (BufferedReader br = new BufferedReader(new FileReader( fileName: "src/main/java/posturasAsana.txt"))) { br: BufferedReader
        String linea; linea: "{"sanskrito": "Parivrtta Trikonasana", "ingles": "Revolved Triangle Pose", "español": "Postura del Triángulo Invertido"}"
        while ((linea = br.readLine()) != null) { br: BufferedReader@2502
            PosturaAsana posturaAsana = gson.fromJson(linea, PosturaAsana.class); linea: "{"sanskrito": "Parivrtta Trikonasana", "ingles": "Revolved Triangle Pose", "español": "Postura del Triángulo Invertido"}"
            this.posturas.add(posturaAsana); posturas: size = 1
        }
    } catch (IOException e) {
        System.out.println("Archivo no encontrado");
    }
}
  
```

**Resultado real y condiciones:** El resultado real constó de la lectura de las líneas del archivo de texto de manera adecuada, independiente del número de líneas que el mismo tiene, en la imagen anterior se apreció en el Debugger el paso a paso del proceso que realiza el método para cometer el acto.

## ISWD652 CALIDAD DE SOFTWARE

### CASO DE PRUEBA 6

ID	Descripción	Supuestos y Condiciones previas
CP06	Se proporciona al método un archivo de texto que contiene n líneas, la lectura de estas deben realizarse en un tiempo que no sea percibido en su totalidad por el usuario.	El programa debe ejecutarse y realizar los pasos adecuados para llegar hasta el método indicado.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El tiempo de ejecución debe ser inferior a un segundo.

```
C:\Users\USER\.jdk\openjdk-20.0.2\bin\java.exe "-java
Tiempo de ejecución del método obtenerPosturas: 0.025
```

**Resultado real y condiciones:** El resultado real constó de una ejecución del método de 0.025 segundos, lo cual a ojo humano es considerado imperceptible.

#### Método obtenerMorfemas()

```
public void obtenerMorfema() {
    try (BufferedReader br = new BufferedReader(new
FileReader("src/main/java/morfemas.txt"))) {
        String linea;
        while ((linea = br.readLine()) != null) {
            Morfema morfemas = gson.fromJson(linea, Morfema.class);
            this.morfemas.add(morfemas);
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

### CASO DE PRUEBA 7

ID	Descripción	Supuestos y Condiciones previas
CP07	Se proporciona al método un archivo de texto válido con datos en un formato correcto	El programa debe estar en ejecución y el archivo dentro de la carpeta raíz del proyecto.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El Arraylist correspondiente debe encontrarse con los datos que se encontraban dentro del archivo de texto proporcionado.

## ISWD652 CALIDAD DE SOFTWARE

```

✓ {..} this = {Diccionario@2578}
  > posturas = {ArrayList@2582} size = 20
  ✓ morfemas = {ArrayList@2580} size = 30
    > {..} 0 = {Morfema@2592} "\nTraducción del MORFEMA: asana\n==> Ingles = posture\n==> Español = postura" ... View
    > {..} 1 = {Morfema@2602} "\nTraducción del MORFEMA: adha\n==> Ingles = down\n==> Español = abajo" ... View
    > {..} 2 = {Morfema@2612} "\nTraducción del MORFEMA: ardh\n==> Ingles = half\n==> Español = mitad" ... View
    > {..} 3 = {Morfema@2622} "\nTraducción del MORFEMA: urdh\n==> Ingles = up\n==> Español = Arriba" ... View
    > {..} 4 = {Morfema@2632} "\nTraducción del MORFEMA: Mukha\n==> Ingles = face\n==> Español = rostro" ... View
    > {..} 5 = {Morfema@2642} "\nTraducción del MORFEMA: Svana\n==> Ingles = dog\n==> Español = perro" ... View
    > {..} 6 = {Morfema@2652} "\nTraducción del MORFEMA: Virabhadra\n==> Ingles = warrior\n==> Español = guerrero" ... View
    > {..} 7 = {Morfema@2662} "\nTraducción del MORFEMA: Vriksha\n==> Ingles = tree\n==> Español = árbol" ... View
    > {..} 8 = {Morfema@2672} "\nTraducción del MORFEMA: Bala\n==> Ingles = kid\n==> Español = niño" ... View
    > {..} 9 = {Morfema@2682} "\nTraducción del MORFEMA: Bhujanga\n==> Ingles = cobra\n==> Español = cobra" ... View
    > {..} 10 = {Morfema@2692} "\nTraducción del MORFEMA: Paschi\n==> Ingles = west\n==> Español = oeste" ... View
    > {..} 11 = {Morfema@2702} "\nTraducción del MORFEMA: Setu\n==> Ingles = bridge\n==> Español = puente" ... View
    > {..} 12 = {Morfema@2712} "\nTraducción del MORFEMA: Shava\n==> Ingles = corpse\n==> Español = cadáver" ... View
    > {..} 13 = {Morfema@2722} "\nTraducción del MORFEMA: Trikona\n==> Ingles = triangle\n==> Español = triángulo" ... View
    > {..} 14 = {Morfema@2732} "\nTraducción del MORFEMA: Matsya\n==> Ingles = fish\n==> Español = pez" ... View
    > {..} 15 = {Morfema@2742} "\nTraducción del MORFEMA: Mala\n==> Ingles = wreath\n==> Español = guirnalda" ... View
    > {..} 16 = {Morfema@2752} "\nTraducción del MORFEMA: Eka\n==> Ingles = one\n==> Español = uno" ... View
    > {..} 17 = {Morfema@2762} "\nTraducción del MORFEMA: Pada\n==> Ingles = toe\n==> Español = pie" ... View
    > {..} 18 = {Morfema@2772} "\nTraducción del MORFEMA: Raja\n==> Ingles = real\n==> Español = real" ... View
    > {..} 19 = {Morfema@2782} "\nTraducción del MORFEMA: Kapota\n==> Ingles = pigeon\n==> Español = paloma" ... View
    > {..} 20 = {Morfema@2792} "\nTraducción del MORFEMA: Hasta\n==> Ingles = hand\n==> Español = mano" ... View
    > {..} 21 = {Morfema@2802} "\nTraducción del MORFEMA: Parivrtta\n==> Ingles = inverted\n==> Español = invertido" ... View
    > {..} 22 = {Morfema@2812} "\nTraducción del MORFEMA: Janu\n==> Ingles = posture\n==> Español = null" ... View
    > {..} 23 = {Morfema@2822} "\nTraducción del MORFEMA: Sirsa\n==> Ingles = posture\n==> Español = null" ... View
    > {..} 24 = {Morfema@2832} "\nTraducción del MORFEMA: Tada\n==> Ingles = mountain\n==> Español = montaña" ... View
    > {..} 25 = {Morfema@2842} "\nTraducción del MORFEMA: Uttana\n==> Ingles = extensive\n==> Español = extenso" ... View
    > {..} 26 = {Morfema@2852} "\nTraducción del MORFEMA: Paschima\n==> Ingles = west\n==> Español = oeste" ... View
  
```

**Resultado real y condiciones:** El resultado real de la ejecución del programa se puede ver en la imagen anterior, en donde se muestra que el arraylist “this.morfemas” se encuentra en efecto con todos los datos encontrados en el txt sin encontrarse error alguno al ejecutar.

### CASO DE PRUEBA 8

ID	Descripción	Supuestos y Condiciones previas
CP08	Se proporciona al método un texto válido pero el mismo se encuentra vacío, entonces, la clase no debe agregar ningún tipo de morfema.	El programa debe estar en ejecución y el archivo dentro de la carpeta raíz del proyecto.
Fecha	Prueba Ejecutada Por	Resultado

## ISWD652 CALIDAD DE SOFTWARE

10-12-2023	Alejandro Jiménez	Aprobado
------------	-------------------	----------

**Resultado esperado:** El arraylist de morfemas debe encontrarse vacío.

```

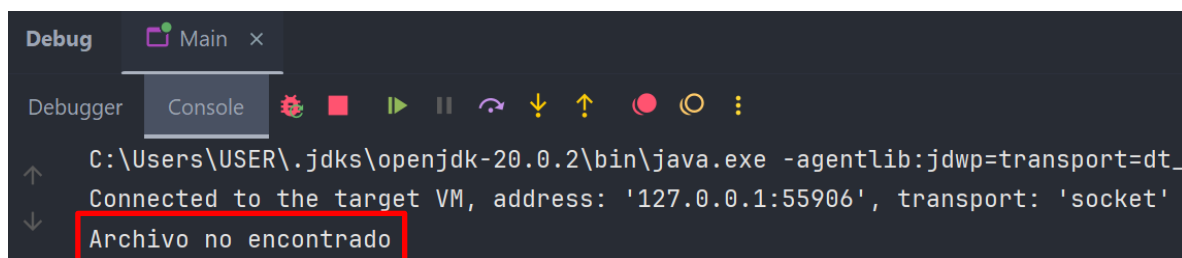
✓ {...} this = {Diccionario@2578}
  > posturas = {ArrayList@2582} size = 20
    morfemas = {ArrayList@2580} size = 0
  
```

**Resultado real y condiciones:** El resultado real muestra en el Debugger como al proporcionar un archivo de texto vacío, simplemente el arraylist se mantiene con longitud 0, es decir, vacío, en lugar de mostrarse cualquier tipo de error.

### CASO DE PRUEBA 9

ID	Descripción	Supuestos y Condiciones previas
CP09	Se proporciona al método un archivo de texto inexistente, entonces, se debería manejar la excepción IOException y la clase no debería verse modificada.	El programa debe estar en ejecución y el archivo no debe existir en la carpeta raíz del proyecto.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debería mostrar en consola un mensaje “Archivo no encontrado” y el arraylist debería seguir vacío.



```

✓ {...} this = {Diccionario@2578}
  > posturas = {ArrayList@2583} size = 20
    morfemas = {ArrayList@2580} size = 0
  
```

**Resultado real y condiciones:** El resultado real fue la impresión del mensaje correspondiente al manejar la excepción, además de no verse modificado el arraylist que se manipulada de existir el archivo de texto.



## ISWD652 CALIDAD DE SOFTWARE

### CASO DE PRUEBA 10

ID	Descripción	Supuestos y Condiciones previas
CP10	Se proporciona al método un archivo de texto el cual consta con una mala estructura para que se extraigan los datos a la clase de manera respectiva, entonces, la clase debe quedar vacía y la excepción controlada.	El programa debe estar en ejecución y el archivo debe existir en la carpeta raíz del proyecto con una estructura inadecuada intencionalmente.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Parcialmente

**Resultado esperado:** El arraylist no debe ingresar los datos proporcionados en dicho archivo de texto, puesto que, la estructura está mal proporcionada, es decir, debería permanecer vacío.

```

{...} this = {Diccionario@2578}
  > posturas = {ArrayList@2583} size = 20
  > morfemas = {ArrayList@2580} size = 1
    {...} 0 = {Morfema@2593} "\nTraducción del MORFEMA: null\n==> Ingles = null\n==> Español = null" ... View
      sanscrito = null
      ingles = null
      español = null

```

**Resultado real y condiciones:** El resultado real fue el añadido de los datos dentro del arraylist, la clase Gson, al no encontrar la estructura adecuada, lo que hizo fue añadir como “null” los campos correspondientes, si bien es cierto, no resultó en un error, sin embargo, puede afectar a la calidad del programa en futuras versiones.

### CASO DE PRUEBA 11

ID	Descripción	Supuestos y Condiciones previas
CP11	Se proporciona al método un archivo de texto que contiene una línea, entonces, la misma debe ser leída y agregada a la clase respectivamente. Debe suceder lo mismo si el archivo contiene n líneas.	El programa debe estar en ejecución y el archivo debe existir en la carpeta raíz del proyecto.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El método debe leer línea por línea al archivo de texto y almacenarla en el arraylist sin errores.



## ISWD652 CALIDAD DE SOFTWARE

```
public void obtenerMorfema() {
    try (BufferedReader br = new BufferedReader(new FileReader( fileName: "src/main/java/morfemas.txt"))) { br: BufferedReader@2586
        String linea; linea: {"sanskrito":"asana","ingles":"posture","español":"postura"}
        while ((linea = br.readLine()) != null) { br: BufferedReader@2586
            Morfema morfemas = gson.fromJson(linea, Morfema.class); linea: {"sanskrito":"asana","ingles":"posture","español":"postura"}
            this.morfemas.add(morfemas); morfemas: size = 1
        }
    } catch (IOException e) {
        System.out.println("Archivo no encontrado");
    }
}
```

**Resultado real y condiciones:** El resultado real constó de la lectura de las líneas del archivo de texto de manera adecuada, independiente del número de líneas que el mismo tiene, en la imagen anterior se apreció en el Debugger el paso a paso del proceso que realiza el método para cometer el acto.

### CASO DE PRUEBA 12

ID	Descripción	Supuestos y Condiciones previas
CP12	Se proporciona al método un archivo de texto que contiene n líneas, la lectura de estas deben realizarse en un tiempo que no sea percibido en su totalidad por el usuario.	El programa debe ejecutarse y realizar los pasos adecuados para llegar hasta el método indicado.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El tiempo de ejecución debe ser inferior a un segundo.

```
C:\Users\USER\.jdk\openjdk-20.0.2\bin\java.exe "-javaa
Tiempo de ejecución del método obtenerMorfem(): 0.003
```

**Resultado real y condiciones:** El resultado real constó de una ejecución del método de 0.003 segundos, lo cual a ojo humano es considerado imperceptible.

### CASO DE PRUEBA 13

ID	Descripción	Supuestos y Condiciones previas
CP13	Se proporciona una postura existente dentro del arraylist de tipo PosturaAsana, entonces, el método to String debe devolver los valores adecuados.	El programa debe ejecutarse y proporcionar la postura que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

## ISWD652 CALIDAD DE SOFTWARE

**Resultado esperado:** Al ingresar una postura previamente almacenada en el programa, el mismo debe devolver su traducción.

Por favor, ingresa la postura en Sánscrito:

La traducción a esa postura es...

Traducción de la POSTURA:  
Paschimottanasana  
==> Inglés: Seated Forward Bend  
==> Español: Flexión hacia adelante sentado

**Resultado real y condiciones:** El resultado real, fue la traducción de la postura en español e inglés de manera adecuada y sin errores.

### CASO DE PRUEBA 14

ID	Descripción	Supuestos y Condiciones previas
CP14	Se proporciona una postura que no existe dentro del arraylist de tipo PosturaAsana, entonces, el método debe devolver el valor "Postura no encontrada"	El programa debe ejecutarse y proporcionar la postura que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debe mostrar en consola "Postura no encontrada" en lugar de un error de ejecución.

Por favor, ingresa la postura en Sánscrito:

La traducción a esa postura es...

Postura no encontrada

**Resultado real y condiciones:** El resultado real fue el esperado, mostrar que dicha postura no existe en el programa con un mensaje.

### CASO DE PRUEBA 15

ID	Descripción	Supuestos y Condiciones previas
CP15	Se proporciona una postura existente dentro del programa, pero se intercala entre mayúsculas y minúsculas, entonces, el programa debe devolver la postura ya que debe ser insensible a las mismas	El programa debe ejecutarse y proporcionar la postura que se desea buscar en el formato indicado para la prueba.

## ISWD652 CALIDAD DE SOFTWARE

	siempre y cuando la ortografía sea la adecuada.	
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** La traducción debe mostrarse en pantalla independientemente de cómo esté escrita siempre y cuando esté bien caligráficamente.

Por favor, ingresa la postura en Sánscrito:

La traducción a esa postura es...

UtkATasANa

Traducción de la POSTURA:  
Utkatasana  
==> Inglés: Chair Pose  
==> Español: Postura de la Silla

**Resultado real y condiciones:** Se presenta la traducción de la postura independientemente de su escritura, destacando que no importa cómo se ingrese siempre y cuando esté bien escrita caligráficamente.

### CASO DE PRUEBA 16

ID	Descripción	Supuestos y Condiciones previas
CP16	Si la lista de posturas se encuentra vacía por x o y motivo, al realizar la búsqueda, no debería marcar "NullPointerException" sino manejarse la excepción.	El programa debe ejecutarse y proporcionar la postura que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** No debe mostrarse ningún error en consola.

```

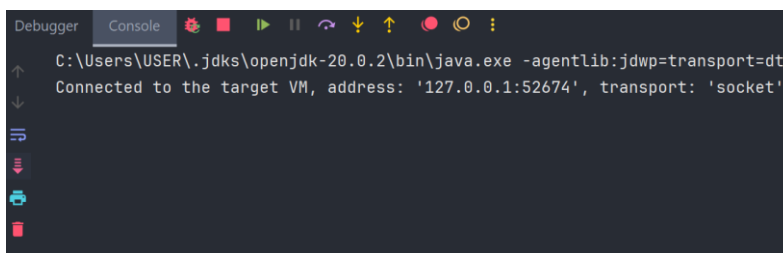
✓ {...} this = {Diccionario@2489}
  posturas = {ArrayList@2491} size = 0

```

La traducción a esa postura es...

Postura no encontrada

## ISWD652 CALIDAD DE SOFTWARE



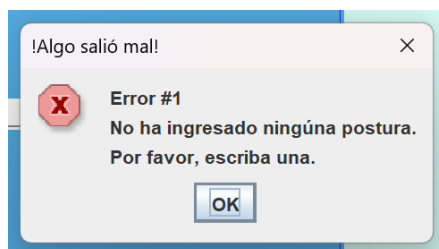
```
Debugger Console
C:\Users\USER\.jdk\openjdk-20.0.2\bin\java.exe -agentlib:jdwp=transport=dt_
Connected to the target VM, address: '127.0.0.1:52674', transport: 'socket'
```

**Resultado real y condiciones:** No se mostró ningún error en consola y se manejó la excepción mostrando “Postura no encontrada”.

### CASO DE PRUEBA 17

ID	Descripción	Supuestos y Condiciones previas
CP17	Si no se ingresa una postura en los campos específicos para el acto, el programa debería advertir al usuario sobre esto.	El programa debe ejecutarse y proporcionar la postura que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Una especie de advertencia debe mostrarse en pantalla.



**Resultado real y condiciones:** Se ha mostrado una advertencia que alerta al usuario que no ha ingresado ninguna postura para su traducción.

### CASO DE PRUEBA 18

ID	Descripción	Supuestos y Condiciones previas
CP18	Si se ingresan números, caracteres especiales o letras no reconocidas por el programa o bien se debería advertir al usuario, o debería saltar la excepción manejada como “Postura no encontrada”	El programa debe ejecutarse y proporcionar la postura que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

## ISWD652 CALIDAD DE SOFTWARE

**Resultado esperado:** Se debería advertir al usuario que no se aceptan caracteres especiales o número o bien manejarse con la excepción.

<p>Por favor, ingresa la postura en Sánscrito:</p> <div style="border: 1px solid black; height: 80px; width: 100%;"></div> <p>!#\$#"</p>	<p>La traducción a esa postura es...</p> <div style="border: 1px solid black; padding: 5px; background-color: #e0ffff;"> <p>Postura no encontrada</p> </div>
--	--

**Resultado real y condiciones:** El resultado real fue el manejo del caso con la excepción del método que obtiene las posturas, mostrando en pantalla “Postura no encontrada”

### CASO DE PRUEBA 19

ID	Descripción	Supuestos y Condiciones previas
CP19	El método de búsqueda implementado no debería demorar más de 1 segundo en mostrar en pantalla los resultados.	El programa debe ejecutarse y proporcionar la postura que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El tiempo de ejecución debe ser inferior a 1 segundo

```
C:\Users\USER\.jdk\openjdk-20
Tiempo de ejecución: 0.012
```

**Resultado real y condiciones:** El resultado real constó de una ejecución del método de 0.012 segundos, lo cual a ojo humano es considerado imperceptible.

### CASO DE PRUEBA 20

ID	Descripción	Supuestos y Condiciones previas
CP20	Se proporciona un morfema existente dentro del arraylist de tipo Morfema, entonces, el método to String debe devolver los valores adecuados.	El programa debe ejecutarse y proporcionar el morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Al ingresar un morfema previamente almacenada en el programa, el mismo debe devolver su traducción.

## ISWD652 CALIDAD DE SOFTWARE

**Por favor, ingresa el morfema en Sánscrito:**

*La traducción de ese morfema es...*

Traducción del MORFEMA: Paschima  
 ==> Ingles = west  
 ==> Español = oeste

**Resultado real y condiciones:** El resultado real, fue la traducción del morfema en español e inglés de manera adecuada y sin errores.

### CASO DE PRUEBA 21

ID	Descripción	Supuestos y Condiciones previas
CP21	Se proporciona un morfema que no existe dentro del arraylist de tipo Morfema, entonces, el método debe devolver el valor "No contamos con ese morfema"	El programa debe ejecutarse y proporcionar el morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debe mostrar en consola "No contamos con ese morfema" en lugar de un error de ejecución.

**Por favor, ingresa el morfema en Sánscrito:**

*La traducción de ese morfema es...*

No contamos con ese morfema

**Resultado real y condiciones:** El resultado real fue el esperado, mostrar que dicho morfema no existe en el programa con un mensaje.

## ISWD652 CALIDAD DE SOFTWARE

### CASO DE PRUEBA 22

ID	Descripción	Supuestos y Condiciones previas
CP22	Se proporciona un morfema existente dentro del programa, pero se intercala entre mayúsculas y minúsculas, entonces, el programa debe devolver el morfema ya que debe ser insensible a las mismas siempre y cuando la ortografía sea la adecuada.	El programa debe ejecutarse y proporcionar el morfema que se desea buscar en el formato indicado para la prueba.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** La traducción debe mostrarse en pantalla independientemente de cómo esté escrita siempre y cuando esté bien caligráficamente.

Por favor, ingresa el morfema en Sánscrito:

La traducción de ese morfema es...

Traducción del MORFEMA: Paschima  
 ==> Ingles = west  
 ==> Español = oeste

**Resultado real y condiciones:** Se presenta la traducción del morfema independientemente de su escritura, destacando que no importa cómo se ingrese siempre y cuando esté bien escrito caligráficamente.

### CASO DE PRUEBA 23

ID	Descripción	Supuestos y Condiciones previas
CP23	Si la lista de morfemas se encuentra vacía por x o y motivo, al realizar la búsqueda, no debería marcar "NullPointerException" sino manejarse la excepción.	El programa debe ejecutarse y proporcionar el morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** No debe mostrarse ningún error en consola.



## ISWD652 CALIDAD DE SOFTWARE

```

{...} this = {Diccionario@2578}
  > posturas = {ArrayList@2583} size = 20
  > morfemas = {ArrayList@2580} size = 0
  
```

Por favor, ingresa el morfema en Sánscrito:

La traducción de ese morfema es...

No contamos con ese morfema

```

Run Main x
C:\Users\USER\.jdk\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains
  
```

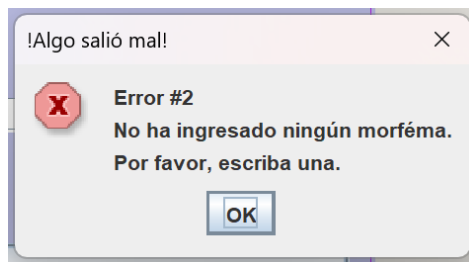
**Resultado real y condiciones:** No se mostró ningún error en consola y se manejó la excepción mostrando “No contamos con ese morfema”.

### CASO DE PRUEBA 24

ID	Descripción	Supuestos y Condiciones previas
CP24	Si no se ingresa un morfema en los campos específicos para el acto, el programa debería advertir al usuario sobre esto.	El programa debe ejecutarse y proporcionar el morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Una especie de advertencia debe mostrarse en pantalla.

## ISWD652 CALIDAD DE SOFTWARE



**Resultado real y condiciones:** Se ha mostrado una advertencia que alerta al usuario que no ha ingresado ningún morfema para su traducción.

### CASO DE PRUEBA 25

ID	Descripción	Supuestos y Condiciones previas
CP25	Si se ingresan números, caracteres especiales o letras no reconocidas por el programa o bien se debería advertir al usuario, o debería saltar la excepción manejada como “No contamos con ese morfema”	El programa debe ejecutarse y proporcionar el morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debería advertir al usuario que no se aceptan caracteres especiales o números o bien manejarse con la excepción.



**Resultado real y condiciones:** El resultado real fue el manejo del caso con la excepción del método que obtiene los morfemas, mostrando en pantalla “No contamos con ese morfema”

## ISWD652 CALIDAD DE SOFTWARE

### CASO DE PRUEBA 26

ID	Descripción	Supuestos y Condiciones previas
CP26	El método de búsqueda implementado no debería demorar más de 1 segundo en mostrar en pantalla los resultados.	El programa debe ejecutarse y proporcionar el morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** El tiempo de ejecución debe ser inferior a 1 segundo

```
C:\Users\USER\.jdk\openjdk-20.0.2\bin\java.exe
Tiempo de ejecucion de la búsqueda: 0.003
```

**Resultado real y condiciones:** El resultado real constó de una ejecución del método de 0.003 segundos, lo cual a ojo humano es considerado imperceptible.

### CASO DE PRUEBA 27

ID	Descripción	Supuestos y Condiciones previas
CP27	En el archivo de texto, se encuentra correctamente ingresadas las palabras bases separadas por “;”, entonces, el programa debe devolver la traducción de cada Morfema encontrado.	El programa debe ejecutarse y proporcionar la postura/morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debe mostrar en pantalla la traducción de la postura con sus morfemas respectivamente.

Por favor, ingresa la postura en Sánscrito:

La traducción a esa postura es...

Traducción de la POSTURA:

Utkatasana

==> Inglés: Chair Pose

==> Español: Postura de la Silla

Traducción del MORFEMA: Utkata

==> Inglés = intense

==> Español = intenso

Traducción del MORFEMA: asana

==> Inglés = posture

==> Español = postura

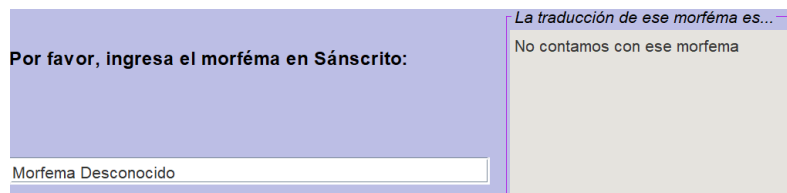
**Resultado real y condiciones:** Se muestra en pantalla la traducción de la postura, adicionalmente se observan los morfemas que la componen junto con su traducción.

## ISWD652 CALIDAD DE SOFTWARE

### CASO DE PRUEBA 28

ID	Descripción	Supuestos y Condiciones previas
CP28	En el caso de encontrarse con morfemas desconocidos en el momento de ejecución, se debería mostrar en pantalla “morfema no encontrado”	El programa debe ejecutarse y proporcionar la postura/morfema que se desea buscar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Al momento de encontrarse con un morfema desconocido debe mostrarse un mensaje que maneje dicho caso.

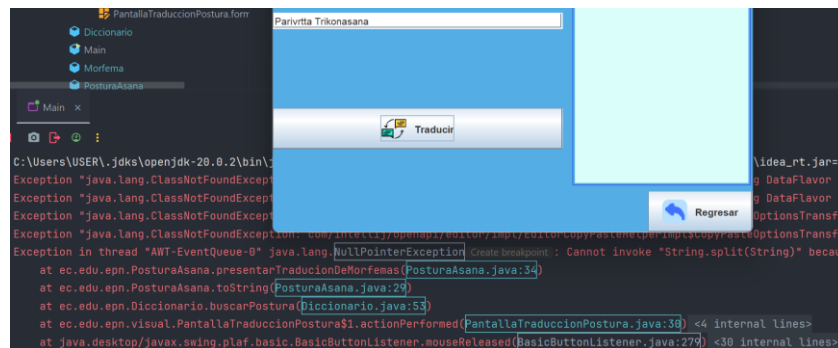


**Resultado real y condiciones:** Se muestra en pantalla que no se cuenta con ese morfema, y no se muestra traducción ni errores en consola.

### CASO DE PRUEBA 29

ID	Descripción	Supuestos y Condiciones previas
CP29	En el caso de que en el archivo de texto no se proporcione palabras base de dicha postura, se debería manejar la excepción.	El programa debe ejecutarse y el archivo de texto debe tener una postura sin palabras base y se debe buscar dicha postura.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Fallida

**Resultado esperado:** Se debería manejar la excepción y no mostrarse ningún error en consola



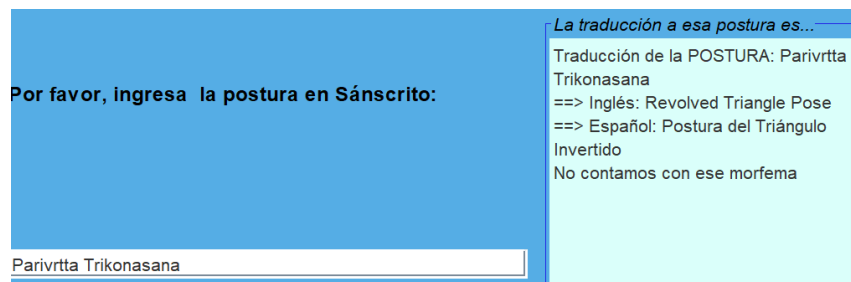
## ISWD652 CALIDAD DE SOFTWARE

**Resultado real y condiciones:** No se ha manejado dicho caso, por lo que, en consola sale el error “NullPointerException”.

### CASO DE PRUEBA 30

ID	Descripción	Supuestos y Condiciones previas
CP30	Si en el archivo de texto, las palabras base no se encuentran separadas por “;” y se encuentran separadas por cualquier otro carácter, el programa debería manejar la excepción e indicar que la sintaxis en las palabras base de la postura x se encuentra mal.	El programa debe ejecutarse y debe existir un archivo de texto con las condiciones expuestas en la descripción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Parcialmente

**Resultado esperado:** Se debería manejar la excepción.



**Resultado real y condiciones:** Se ha manejado la excepción mediante el mensaje “no contamos con ese morfema”, sin embargo, no se avisa al usuario sobre la sintaxis incorrecta dentro del archivo de texto.

### CASO DE PRUEBA 31

ID	Descripción	Supuestos y Condiciones previas
CP31	Al momento de ejecutar el programa se debe mostrar la pantalla de bienvenida.	El programa debe ejecutarse.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** La pantalla de bienvenida debe mostrarse.

## ISWD652 CALIDAD DE SOFTWARE



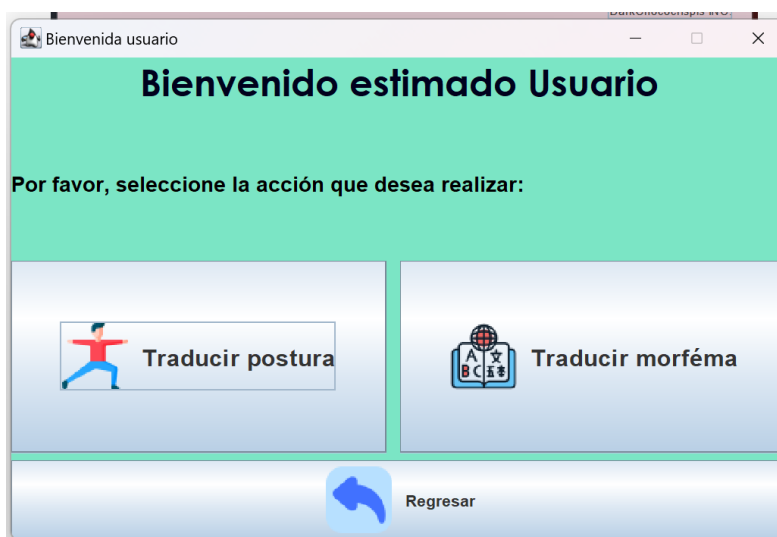
**Resultado real y condiciones:** Se muestra la pantalla de bienvenida sin errores de ejecución.

### CASO DE PRUEBA 32

ID	Descripción	Supuestos y Condiciones previas
CP32	Al momento de ejecutar el programa se debe mostrar la pantalla de bienvenida, luego, al dar clic sobre el botón “iniciar” se debe mostrar la pantalla de opciones de traducción.	El programa debe ejecutarse y debe darse clic sobre el botón.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** La pantalla de traducciones se debe mostrar.

## ISWD652 CALIDAD DE SOFTWARE



**Resultado real y condiciones:** Se muestra la pantalla de traducciones sin errores de ejecución.

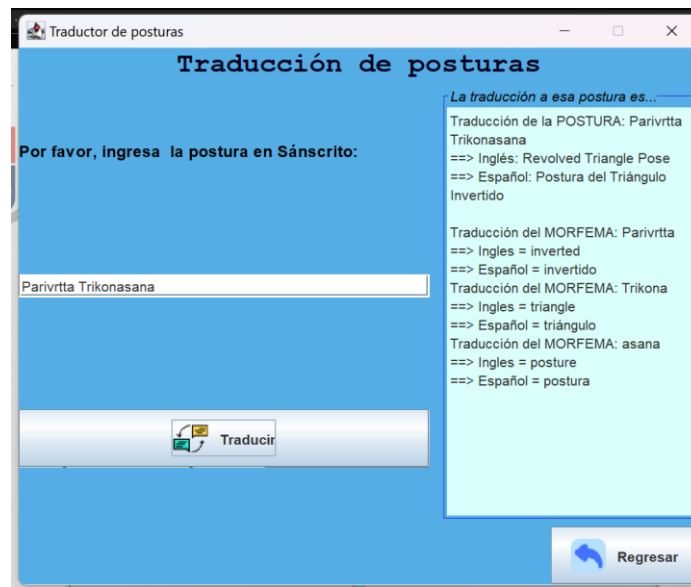
### CASO DE PRUEBA 33

ID	Descripción	Supuestos y Condiciones previas
CP33	Al momento de ejecutar el programa se debe mostrar la pantalla de bienvenida, luego, al dar clic sobre el botón “iniciar” se debe mostrar la pantalla de opciones de traducción, luego, al dar clic sobre cualquiera de los dos botones de traducción, se debe mostrar la pantalla del traductor, además, al ingresar el campo correspondiente y dar clic sobre el botón traducir debe ejecutarse sin problema.	El programa debe ejecutarse y debe darse clic sobre los botones correspondientes y llenar el campo.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debe mostrar la traducción de la postura sin errores al dar clic sobre el botón traducir.



## ISWD652 CALIDAD DE SOFTWARE



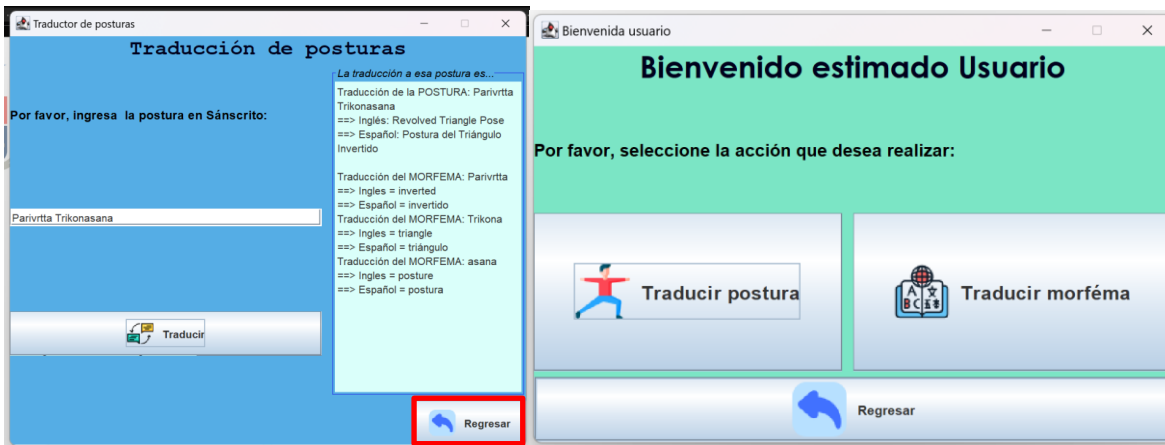
**Resultado real y condiciones:** Se muestra la pantalla del traductor y la traducción del campo indicado sin error o inconveniente alguno.

### CASO DE PRUEBA 34

ID	Descripción	Supuestos y Condiciones previas
CP34	Al dar clic sobre cualquiera de los botones “regresar” encontrados a lo largo de la ejecución del programa, se debe observar la pantalla anteriormente abierta.	El programa debe ejecutarse y debe darse clic sobre el botón regresar.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debe navegar entre pantallas con los botones de manera adecuada.

## ISWD652 CALIDAD DE SOFTWARE



**Resultado real y condiciones:** Se navega entre pantallas sin mostrar errores en consola.

### CASO DE PRUEBA 35

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de un morfema de manera correcta.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Aparecerá la traducción del morfema en pantalla.



## ISWD652 CALIDAD DE SOFTWARE

**Resultado real y condiciones:** Se mostró la traducción del morfema tal y como se esperaba y muestra la imagen anterior.

### CASO DE PRUEBA 36

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de un morfema de manera correcta con espacios en blanco, al final, al inicio, entre la palabra.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Fallido

**Resultado esperado:** Se debe mostrar la traducción del morfema.



**Resultado real y condiciones:** Se descubre que el programa no es sensible a espaciados en blanco al inicio, final o entre la palabra ya que no encuentra al morfema pese a que está bien escrito.

### CASO DE PRUEBA 37

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de un morfema y lo escribe todo con mayúsculas.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debe mostrar la traducción del morfema sin mayor inconveniente.

## ISWD652 CALIDAD DE SOFTWARE

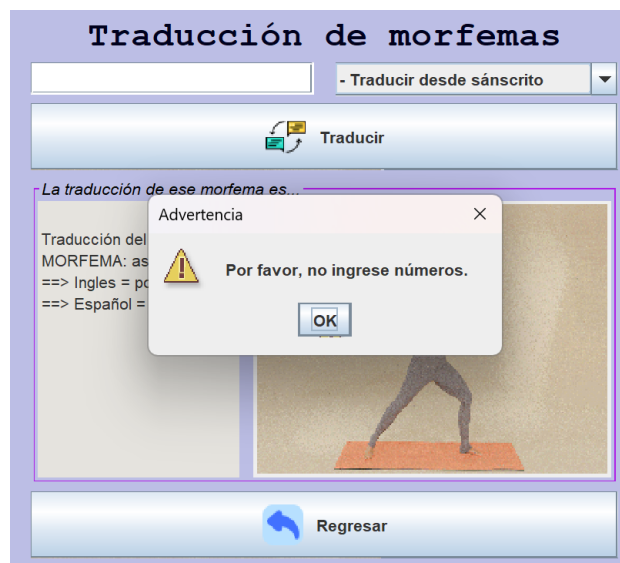


**Resultado real y condiciones:** El resultado real es el esperado, mostrándose así la traducción del morfema.

### CASO DE PRUEBA 38

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de un morfema e ingresa números.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Parcialmente

**Resultado esperado:** Debe dar aviso de que no se pueden ingresar números.



## ISWD652 CALIDAD DE SOFTWARE

**Resultado real y condiciones:** Pese a que no es el caso de la prueba, se ha descubierto que al ingresar números, sale el aviso pero la pantalla no se limpia.

### CASO DE PRUEBA 39

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de una postura de manera correcta.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Aparecerá la traducción de la postura en pantalla.



**Resultado real y condiciones:** Se mostró la traducción de la postura tal y como se esperaba y muestra la imagen anterior.

### CASO DE PRUEBA 36

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de un postura de manera correcta con espacios en blanco, al final, al inicio, entre la palabra.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Fallido

**Resultado esperado:** Se debe mostrar la traducción de la postura.

## ISWD652 CALIDAD DE SOFTWARE



**Resultado real y condiciones:** Se descubre que el programa no es sensible a espaciados en blanco al inicio, final o entre la palabra ya que no encuentra la postura pese a que está bien escrito.

### CASO DE PRUEBA 37

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de un postura y lo escribe todo con mayúsculas.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Aprobado

**Resultado esperado:** Se debe mostrar la traducción de la postura sin mayor inconveniente.



## ISWD652 CALIDAD DE SOFTWARE

**Resultado real y condiciones:** El resultado real es el esperado, mostrándose así la traducción de la postura.

### CASO DE PRUEBA 38

ID	Descripción	Supuestos y Condiciones previas
CP34	Realizar una consulta de un postura e ingresa números.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Parcialmente

**Resultado esperado:** Debe dar aviso de que no se pueden ingresar números.



**Resultado real y condiciones:** Pese a que no es el caso de la prueba, se ha descubierto que, al ingresar números, sale el aviso, pero la pantalla no se limpia.

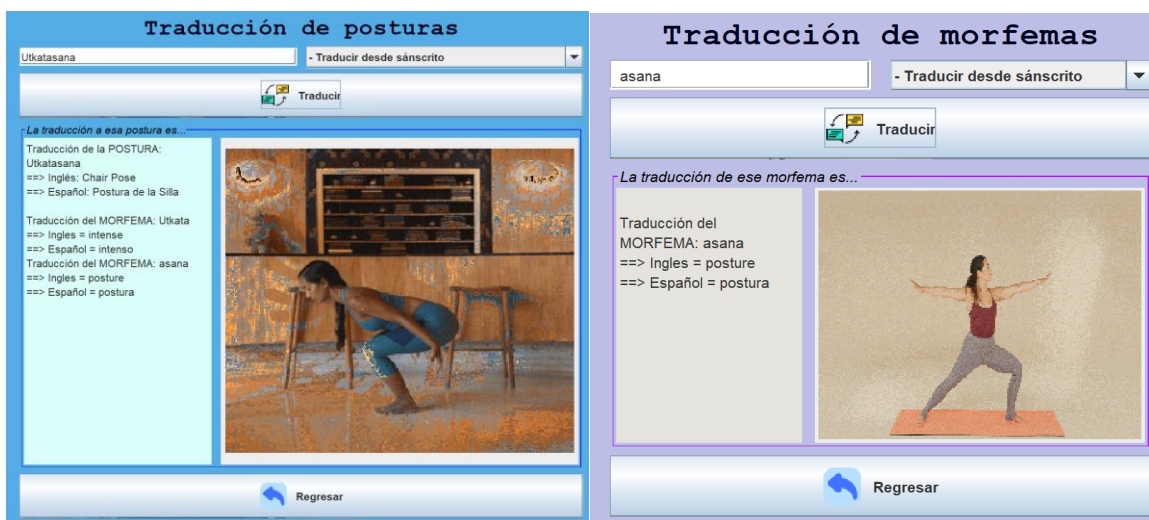
### CASO DE PRUEBA 39

ID	Descripción	Supuestos y Condiciones previas
CP34	Al realizar un consulta para traducir una postura o un morfema, el mismo debe aparecer acompañado de una imagen ilustrativa.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Cumplida



## ISWD652 CALIDAD DE SOFTWARE

**Resultado esperado:** Se debe mostrar tanto la traducción del morfema/postura como la imagen ilustrativa.



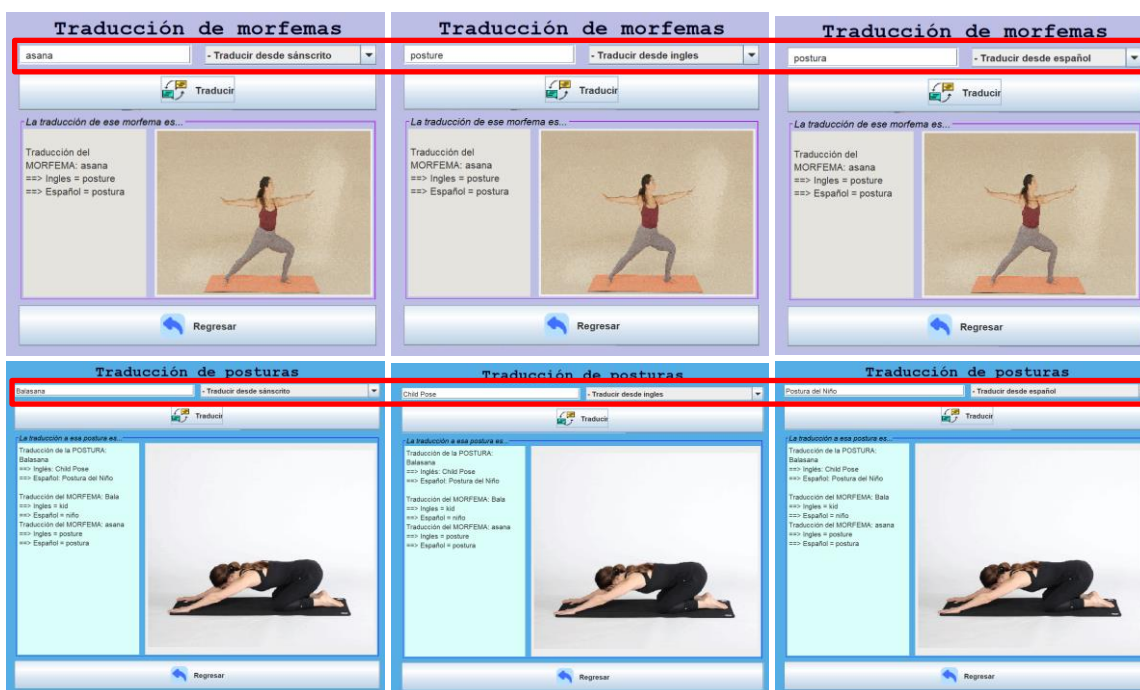
**Resultado real y condiciones:** Se muestra la traducción junto con su imagen ilustrativa de manera adecuada.

### CASO DE PRUEBA 40

ID	Descripción	Supuestos y Condiciones previas
CP34	El programa debe ser capaz de traducir desde sánscrito, español e inglés respectivamente, mostrando el mismo mensaje para cada ocasión sea de un morfema o postura.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
<b>Fecha</b>	<b>Prueba Ejecutada Por</b>	<b>Resultado</b>
10-12-2023	Alejandro Jiménez	Cumplida

**Resultado esperado:** Se debe mostrar la traducción correspondiente, siempre debe ser la misma independientemente del idioma que se elija.

## ISWD652 CALIDAD DE SOFTWARE



**Resultado real y condiciones:** Las traducciones funcionan tanto para sanscrito, español e inglés.

### CASO DE PRUEBA 41

ID	Descripción	Supuestos y Condiciones previas
CP34	Si se ingresa la postura en inglés, pero la aplicación se encuentra en traducción de sánscrito debería o notificar que ese no es el idioma o a su vez no encontrar la postura.	El programa debe ejecutarse y el usuario debe estar en la ventana correspondiente a la traducción.
Fecha	Prueba Ejecutada Por	Resultado
10-12-2023	Alejandro Jiménez	Cumplida

**Resultado esperado:** Se debe avisar de alguna forma al usuario sobre el error que está cometiendo.

## ISWD652 CALIDAD DE SOFTWARE



**Resultado real y condiciones:** Pese a que de alguna forma se da aviso, originalmente, no es que no se haya encontrado la postura, más bien, el usuario a ingresado un idioma al que no corresponde la traducción.

### LISTA DE DEFECTOS DE CALIDAD

Nº	Tipo	Descripción
1	Funcional - Chequeo	No existe un control en la lectura de datos de tipo "null" en los archivos de texto.
2	Funcional - Chequeo	No existe una verificación de que los datos de las posturas tengan sus respectivas palabras base.
3	Funcional - Chequeo	Al no existir ninguna postura que emplee números, debería haber un control para que el usuario no las ingrese.
4	No funcional - Documentación	El código no posee documentación adecuada.