

## Manual Técnico Sopa de Letras

### Metodo MenúPrincipal()

Este metodo es el Menú principal del programa en el cual Tenemos 7 opciones, las cuales son:

```
1. NUEVA PARTIDA
2. HISTORIAL DE PARTIDAS
3. MOSTRAR PUNTUACIONES MAS ALTAS
4. MOSTRAR JUGADORES QUE NO ENCONTRARON TODAS LAS PALABRAS
5. MOSTRAR JUGADORES QUE ENCONTRARON TODAS LAS PALABRAS
6. MOSTRAR INFORMACION DEL ESTUDIANTE
7. SALIR
```

```
*****
Ingrese una opción:
*****
```

Es la primera interacción con el usuario, n el cual debe elegir una opción para continuar con la ejecución del programa.

Al entrar a nueva partida se le pedirá el nombre y el tamaño del tablero.

### Metodo MenúJuego()

```
1. MENU PALABRAS
2. JUGAR
3. TERMINAR PARTIDA
*****
INGRESE UNA OPCION:
*****
```

En este menú el usuario podrá elegir entre entrar al menú de palabras, empezar a jugar o terminar la partida. Al terminar la partida regresará al menú principal.

### Metodo SubMenúJuego()

```
1. INSERTAR
2. MODIFICAR
3. ELIMINAR
4. SALIR
*****
INGRESE UNA OPCION
*****
```

Este metodo se muestra el menú para que el usuario elija entre insertar palabras, modificar las palabras o eliminar las palabras. Al salir regresará al Menú de nueva partida.

### Metodo ordenarHorizontal()

Este metodo es utilizado para insertar la palabra en horizontal en el arreglo.

### **Metodo ordenarVertical()**

Este metodo es utilizado para insertar la palabra en vertical en el arreglo.

### **Metodo VerSopa()**

Este metodo sirve para visualizar el arreglo.

### **Metodo LLenarSopa()**

Este metodo sirve para llenar los espacios vacios de la matriz con letras al azar

### **Metodo Insertar()**

Este metodo es utilizado para guardar las palabras dependiendo de cuantas quiere ingresar el usuario en un rango de 1 a 10 palabras y las guardar en una variable.

Las palabras pueden tener un rango de 5 a 10 caracteres.

### **Metodo Modificar()**

Este metodo permite modificar las palabras, verifica que se le haya sido asignado un valor y modifica la palabra.

### **Metodo Eliminar()**

Nos permite eliminar una de las palabras que teníamos ingresadas.

### **Metodo ModificarPalabra1()**

Si la palabra que deseábamos modificar es la 1 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

### **Metodo ModificarPalabra2()**

Si la palabra que deseábamos modificar es la 2 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

### **Metodo ModificarPalabra3()**

Si la palabra que deseábamos modificar es la 3 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

### **Metodo ModificarPalabra4()**

Si la palabra que deseábamos modificar es la 4 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

### **Metodo ModificarPalabra5()**

Si la palabra que deseábamos modificar es la 5 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

#### **Metodo ModificarPalabra6()**

Si la palabra que deseábamos modificar es la 6 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

#### **Metodo ModificarPalabra7()**

Si la palabra que deseábamos modificar es la 7 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

#### **Metodo ModificarPalabra8()**

Si la palabra que deseábamos modificar es la 8 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

#### **Metodo ModificarPalabra9()**

Si la palabra que deseábamos modificar es la 9 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

#### **Metodo ModificarPalabra10()**

Si la palabra que deseábamos modificar es la 10 pide la palabra para remplazarla y verifica que tenga de 5 a 10caracteres.

#### **Metodo PalabraVacia1()**

Verifica que la palabra 1 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

#### **Metodo PalabraVacia2()**

Verifica que la palabra 2 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

#### **Metodo PalabraVacia3()**

Verifica que la palabra 3 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

#### **Metodo PalabraVacia4()**

Verifica que la palabra 4 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

#### **Metodo PalabraVacia5()**

Verifica que la palabra 5 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

#### **Metodo PalabraVacia6()**

Verifica que la palabra 6 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

#### **Metodo PalabraVacia7()**

Verifica que la palabra 7 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

#### **Metodo PalabraVacia8()**

Verifica que la palabra 8 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

**Metodo PalabraVacía9()**

Verifica que la palabra 9 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

**Metodo PalabraVacía10()**

Verifica que la palabra 10 ingresada tenga un valor de lo contrario lo regresa al menú de palabras.

**Metodo MatrizPalabra1(String palabra)**

Este metodo sirve para asignar de manera aleatoria si la palabra que ingresamos va a ser ingresada de manera horizontal o vertical. Tambien nos dice en que fila y columna será ingresada la palabra de manera aleatoria e ingresa la palabra en la matriz.

**Metodo MatrizDefecto1()**

Ingresa la palabra por defecto1 de manera aleatoria ya sea vertical u horizontal, y en una posición aleatoria.

**Metodo MatrizDefecto2()**

Ingresa la palabra por defecto2 de manera aleatoria ya sea vertical u horizontal, y en una posición aleatoria.

**Metodo MatrizDefecto3()**

Ingresa la palabra por defecto3 de manera aleatoria ya sea vertical u horizontal, y en una posición aleatoria.

**Metodo MatrizDefecto4()**

Ingresa la palabra por defecto4 de manera aleatoria ya sea vertical u horizontal, y en una posición aleatoria.

**Metodo MatrizDefecto5()**

Ingresa la palabra por defecto5 de manera aleatoria ya sea vertical u horizontal, y en una posición aleatoria.

**Metodo encontrarPalabra()**

Este metodo nos sirve para verificar si la palabra que el usuario encontró en la sopa de letras, existe en el arreglo. Si existe nos pide la posición de dicha palabra y si es correcto cambia la palabra por \$. En este metodo tambien nos ayuda a llevar el punteo y las vidas del jugador, si este se equivoca de palabra se le restara 1 vida y 5 puntos. Si acierta se le sumaran los puntos de la longitud de la palabra encontrada.