

Manual Técnico: Generador de Página Web en Fortran

1. Descripción General del Proyecto

El proyecto Generador de Página Web en Fortran analiza un archivo de entrada de texto que contiene definiciones de controles, propiedades y ubicaciones para elementos de una página web. El programa realiza un análisis léxico y genera automáticamente un archivo HTML y un archivo CSS a partir de estas definiciones.

2. Requisitos del Proyecto

Software:

- **Compilador Fortran:** Se requiere `gfortran` o un compilador similar para compilar el programa.
- **Editor de Texto:** Para crear y modificar el archivo de entrada (ej. VS Code, Notepad).
- **Navegador Web:** Para visualizar el resultado en formato HTML.

Hardware:

- **Procesador:** CPU de 1 GHz o superior.
- **Memoria:** 512 MB de RAM o superior.
- **Espacio en Disco:** Aproximadamente 1 MB para archivos del programa y almacenamiento temporal.

3. Estructura del Proyecto

- **entrada.txt:** Archivo de entrada con la definición de los controles, propiedades y ubicaciones.
- **analizador.f90:** Código fuente en Fortran que analiza `entrada.txt` y genera los archivos de salida.
- **salida.html:** Archivo HTML generado, contiene la estructura visual de la página.
- **estilos.css:** Archivo CSS generado, contiene los estilos aplicados a los elementos HTML.

4. Arquitectura del Programa

El programa se compone de tres módulos principales, implementados como subrutinas en el código Fortran:

1. **Análisis Léxico:** `analizar_linea` detecta y clasifica cada línea del archivo de entrada en una de las tres categorías (controles, propiedades, colocación). Ignora comentarios y maneja errores léxicos.
2. **Generación HTML:** `procesar_control` crea las etiquetas HTML para los controles definidos.

3. **Generación CSS:** `procesar_propiedad` y `procesar_colocacion` generan el archivo CSS aplicando estilos y posiciones.

5. Descripción de Funciones y Subrutinas

`program analizador_html_css`

Controla la ejecución principal del programa. Abre los archivos de entrada y salida, recorre cada línea del archivo de entrada, y utiliza las subrutinas para generar los archivos HTML y CSS.

`subroutine analizar_linea`

Clasifica cada línea del archivo de entrada:

- **Controles:** Crea elementos HTML.
- **Propiedades:** Aplica estilos CSS.
- **Colocación:** Define la ubicación de los elementos en CSS.

`subroutine procesar_control`

Genera etiquetas HTML en función de los controles definidos en el archivo de entrada. Por ejemplo:

- Contenedor -> `<div>`
- Boton -> `<input type="button">`

`subroutine procesar_propiedad`

Genera estilos CSS para los controles. Asocia propiedades como el color de fondo (`setColorFondo`), color de letra (`setColorLetra`), y tamaño (`setAncho` y `setAlto`).

`subroutine procesar_colocacion`

Define las posiciones de los elementos usando CSS, permitiendo posicionar controles mediante coordenadas (por ejemplo, `setPosicion`), márgenes y alineaciones.

6. Flujo de Ejecución

1. **Compilación:**

- Compila el archivo con:

```
bash
Copiar código
gfortran -o analizador analizador.f90
```

2. **Ejecución:**

- Ejecuta el programa con:

```
bash
Copiar código
./analizador
```

3. Entrada y Salida:

- El programa lee `entrada.txt`, clasifica y genera los archivos `salida.html` y `estilos.css`.

7. Manejo de Errores

El programa implementa controles de errores en los siguientes puntos:

1. **Apertura de Archivos:** Si el archivo de entrada no existe o no se puede abrir, el programa muestra un mensaje de error y se detiene.
2. **Errores Léxicos:** Las líneas de entrada que no cumplen con las expectativas de formato (por ejemplo, propiedades o colocaciones mal escritas) son reportadas en la consola como errores léxicos.
3. **Comentarios y Bloques:** Ignora comentarios de una línea (`//`) y de varias líneas (`/* ... */`) para evitar procesar información no deseada.

8. Formato del Archivo de Entrada (`entrada.txt`)

El archivo de entrada tiene tres secciones delimitadas:

```
plaintext
Copiar código
<!--Controles
Control ID;
...
Controles -->

<!--Propiedades
ID.setPropiedad(valor);
...
Propiedades -->

<!--Colocacion
ID.setPosicion(x,y);
...
Colocacion -->
```

9. Ejemplo de Entrada y Resultados Esperados

Ejemplo de `entrada.txt`

```
plaintext
Copiar código
<!--Controles
```

```

Contenedor contlogin;
Boton cmdIngresar;
Etiqueta lblUsuario;
Controles -->

<!--Propiedades
contlogin.setAncho(400);
contlogin.setColorFondo(240,240,240);
lblUsuario.setColorLetra(0,0,255);
cmdIngresar.setTexto("Ingresar");
Propiedades -->

<!--Colocacion
contlogin.setPosicion(50,100);
lblUsuario.setPosicion(70,120);
cmdIngresar.setPosicion(200,150);
Colocacion -->

```

Resultados Esperados

- **HTML** (salida.html):

```

html
Copiar código
<!DOCTYPE html>
<html>
<head>
    <link rel='stylesheet' type='text/css' href='estilos.css'>
    <title>Formulario Generado</title>
</head>
<body>
    <div class='container'>
        <h1>Formulario de Controles</h1>
        <div id='contlogin'></div>
        <label id='lblUsuario'>Texto de la Etiqueta</label>
        <input type='button' id='cmdIngresar' value='Ingresar'
class='boton' />
    </div>
</body>
</html>

```

- **CSS** (estilos.css):

```

css
Copiar código
body { font-family: Arial, sans-serif; background-color: #f4f4f4;
padding: 20px; }
.container { width: 80%; margin: 0 auto; background-color: #fff;
padding: 20px; border-radius: 8px; box-shadow: 0px 0px 10px rgba(0,
0, 0, 0.1); }
h1 { text-align: center; color: #333; }
#contlogin { width: 400px; background-color: rgb(240,240,240); }
#lblUsuario { color: rgb(0,0,255); }
#cmdIngresar { text-align: center; }

```

10. Mantenimiento y Extensión del Código

1. **Añadir Nuevos Controles:** Para añadir un nuevo tipo de control, edita `procesar_control` e incluye una nueva condición para ese control.
2. **Propiedades y Estilos Adicionales:** Agrega nuevas propiedades y estilos en `procesar_propiedad`, especificando el CSS correspondiente.
3. **Soporte para Posiciones Avanzadas:** Extiende `procesar_colocacion` para admitir posiciones avanzadas como `float`, `flexbox`, etc.

11. Contacto y Documentación

Para soporte técnico o futuras actualizaciones, contacta al desarrollador a través del canal de soporte correspondiente.