

UNIVERSIDAD SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

Organización Lenguajes y Compiladores 1

Escuela de vacaciones

Diciembre de 2018

Ing. Mario Bautista

Tutor Académico

Luis Paz



PROYECTO 1: GAME-COMPILER

OBJETIVOS

OBJETIVOS GENERALES

- Que el estudiante aplique los conocimientos aprendidos en el curso de Compiladores 1 y Lenguajes Formales de Programación.
- Que el estudiante implemente las fases de un compilador en el análisis léxico y análisis sintáctico.
- Que el estudiante comprenda y utilice la herramienta Irony.

OBJETIVOS ESPECIFICOS

- Aplicar el concepto de compiladores como alternativa de solución de problemas.
- Que el estudiante realice una aplicación interactiva que trabaje conjuntamente con el analizador.
- Que el estudiante sea capaz de crear gramáticas y de recuperarse de errores léxicos, sintácticos.
- Iniciar el proceso de aprendizaje de herramientas generadoras de Analizadores Léxicos y Sintácticos.

INTRODUCCION

La empresa GameSolution necesita crear una herramienta capaz de realizar diferentes escenarios o niveles, para su más novedoso videojuego, para facilitar su uso y que sea agradable, útil y distractor al usuario final.

Se necesita que a través de un analizador sintáctico se puedan crear objetos y personajes para la configuración del videojuego, así también crear escenarios con el fin de lograr probar o ejecutar el escenario con los personajes y objetos configurados.

También debe de proveer una interfaz gráfica amigable al usuario para crear escenarios utilizando los objetos y personajes cargados por el analizador.

DESCRIPCION DEL PROBLEMA

Se necesita que se desarrolle una aplicación que a partir de un lenguaje de programación se realice la creación de escenarios y facilite la construcción del video juego.

Para lo cual se ingresaran dos archivos en forma ordena y la compilación de los mismos nos permitirá ver el juego en una interfaz gráfica.

PRIMER ARCHIVO

ARCHIVO DE CONFIGURACION

Este archivo contiene toda la información sobre las propiedades de los diferentes fondos de escenarios, propiedades de personajes, atributos de objetos. Este archivo se deberá cargar al iniciar la aplicación y podrá ser actualizado por otro archivo con el mismo formato que agregara nueva información.

1. Fondo de Escenario: Es una imagen que se coloca sobre el área del escenario.
2. Héroe y Enemigos: Son los personajes que participaran en el juego, el héroe será el personaje principal del juego que será manipulado por el usuario, en cuanto a los Enemigos estos actuaran de forma automática y estarán con el objetivo de quitarle vida al héroe o matarlo.
3. Escenario: Esta área será utilizada para construir nuestra plataforma del juego, entre los objetos que serán cargados al escenario tenemos.
 - a. Meta: Es el objetivo del jugador, cuando el héroe llegue hasta la meta se finaliza de forma satisfactoria el juego mostrándole un mensaje, es representado por una puerta o salida utilizando una imagen acorde a esta.
 - b. Objetos de Destrucción: Estas pueden ser armas que serán de ayuda al héroe para eliminar a los enemigos, y los objetos bomba serán posicionados para que el héroe pierda vida o energía.
 - c. Bonus: Este objeto proporcionara vida o energía al personaje principal.
 - d. Bloques: los bloques serán utilizados para crear paredes en toda el área del escenario esto con el fin de tener un laberinto para hacer más complejo al héroe buscar la salida.

SEGUNDO ARCHIVO

ARCHIVO DE CONFIGURACION DE LOS ESCENARIOS

Este archivo contiene la ubicación de los personajes, objetos, paredes. Después de cargar se podrá visualizar el escenario, posteriormente podremos hacer un test del escenario al probar o ejecutar el escenario.

DESCRIPCION GRAFICA



PRIMER ARCHIVO

ARCHIVO DE CONFIGURACION

ESTRUCUTRA DE CONFIGURACION

Este archivo se cargara al inicio de la aplicación, se puede cargar otros archivos para añadir elementos, cada etiqueta o atributo en este archivo puede venir en cualquier orden.

<configuration>

<background>

//propiedades para el fondo de pantalla

</background>

<figure>

//propiedades para los personajes

</figure>

<design>

//propiedades para los objetos o elementos

</design>

</configuration>

ATRIBUTOS PRINCIPALE S

x-nombre	Identificador único del personaje u objeto cada objeto debe tener uno en caso contrario se debe descartar. Iniciando con una letra seguida de letra o número o guion bajo.
x-imagen	Ubicación o dirección de la imagen, esta dirección estará encerrada entre comillas dobles, la dirección común mente se conoce como path para la cual deben de hacer una expresión regular que valide que sea una dirección real por ejemplo “/home/usuario/imagenes/fondo1.jpg” o en el caso de Windows “C:\Users\usuario\imagenes\fondo1.jpg”
x-tipo	Para cada elemento, pueden existir múltiples tipos, por ello esta nos ayudara a descartar algunas propiedades del objeto.

FONDO DEL ESCENARIO ETIQUETA BACKGROUN

Es una imagen que se coloca sobre el área del escenario, esta se colocara en la primera posición del escenario y esta se ajustara al tamaño de la pantalla.

En el archivo de configuración puede venir n-elementos de fondo de escenario, para posteriormente ser seleccionado un elemento a usar en el juego.

<background>

```
{  
    x-nombre=fondo_1;  
    x-imagen="/home/usuario/imágenes/fondo1.jpg";  
},  
{  
    x-nombre=fondo_2;  
    x-imagen="/home/usuario/imágenes/fondo2.jpg";  
}
```

</background>

PERSONAJES ETIQUETA FIGURA

Son cada una de las figuras, ya sea héroe o enemigo que aparecen en el juego que tiene diferentes atributos, para su configuración.

En el archivo de configuración puede venir n-figuras de personajes tanto héroes como villanos, para posteriormente ser seleccionadas.

Atributo	Descripción
x-vida	<p>Tipo: Enterotendrán un rango de 0 a 100 [0-100] donde 0 es el valor donde se da por muerto al héroe y 100 es el valor máximo de vida.</p> <p>Indica la cantidad de vida que tiene el personaje, esta puede aumentar al sumar un objeto de bonus o disminuir al ser atacado por un enemigo o un objeto bomba.</p> <p>De no existir este atributo vendrá un valor por defecto de vida igual a 100.</p>

	<p>La vida será aumentada o disminuida según el daño que cause el enemigo o las trampas del escenario.</p>
x-tipo	<p>Tipo: x-heroe x-enemigo</p> <p>Se podrán crear n-personajes de cualquier tipo, estos pueden ser héroes o enemigos.</p> <p>El tipo héroe se podrá declarar más de una vez, pero solo se podrá jugar con un único héroe la partida del juego.</p> <p>El tipo enemigo serán los rivales del usuario que le quitaran vida o lo podrán matar, vendrán varios enemigos y se pueden seleccionar varios enemigos para el nivel determinado.</p> <p>El héroe será movilizado por el usuario utilizando las flechas correspondientes del teclado y el enemigo se moverá automáticamente en un rango de casillas esto con el fin de darle movimiento utilizando inteligencia artificial.</p>
x-destruir	<p>Tipo: Entero</p> <p>Esto indicara la cantidad de destrucción que provocara al héroe sobre la vida.</p> <p>Este atributo solo vendrá al declarar a un enemigo.</p> <p>El rango de destrucción del enemigo será un numero entero de 1 0 100;</p>
x-descripción	<p>Información descriptiva del personaje, que puede ayudar a escoger el personaje al usuario, o a describirlo si el personaje es de agua, tierra, fuego, etc.</p> <p>Opcional de no existir el usuario asignara una descripción automática.</p>

<figure>

```
{  
    x-nombre=MarioBros;  
    x-vida=100;  
    x-imagen="/home/usuario/imágenes/mario.jpg";  
    x-tipo=x-heroe;  
    x-descripcion="Personaje principal del juego el cual tiene como objetivo salvar a la reyna."  
},  
{  
    x-nombre=koopas;  
    x-vida=100;  
    x-imagen="/home/usuario/imágenes/koopas.jpg";  
    x-tipo=x-enemigo;  
    x-destruir=50;  
    x-descripcion="Enemigo principal de mario bros";  
}
```

</figure>

DISEÑO ETIQUETA DESIGN

Existen diferentes objetos para su configuración entre ellos:

Destrucción: Estos objetos ayudaran al héroe o personaje principal a destruir a sus enemigos (armas) como también existirán trampas (bombas) que provocarán que el pierda vida.

Bonus: Estos elementos darán vida a nuestro personaje principal.

Paredes (Bloques): Para que sea más complicado llegar a su destino deberá existir paredes formando laberintos para llegar a la meta.

Meta: Representara el final del juego mostrando un mensaje de éxito.

Los elementos de destrucción, bonus y armas solo se podrán utilizar una vez luego desaparecerán.

Atributo	Descripción
x-tipo	x-meta x-bloque x-bonus x-bomba x-arma x-meta: Representa el final del juego, si el jugador principal llega hasta ella finalizara el juego, se pueden crear múltiples metas pero únicamente se podrá seleccionar 1 al crear el escenario.

	<p>x-bloque: Esto representan las paredes del escenario que es necesario para darle complejidad al jugador de llegar a la meta.</p> <p>x-bomba: representara una trampa para el jugador si el héroe toca el objeto le restara vida y el objeto bomba desaparecerá.</p> <p>x-arma: este objeto ayudara al héroe a matar a los enemigos contara con un numero el cual le será restado a la vida del enemigo.</p> <p>x-bonus: brindara vida al héroe solo se podrá utilizar una única vez luego desaparecerá.</p>
x-destruir	<p>Tipo: Entero</p> <p>Este indica la cantidad de destrucción que provocara a otro personaje sobre la vida, este atributo vendrá únicamente en las armas y las bombas.</p>
x-creditos	Tipo: Entero, únicamente existe bajo la etiqueta x-bonus
x-imagen	Tendra una dirección (path) para el objeto lo cual lo ara más atractivo al juego.

<design>

```

{
    x-nombre=bomba1;
    x-destruir=20;
    x-imagen="/home/usuario/imágenes/bomba1.jpg";
    x-tipo=x-bomba;
},
{
    x-nombre=arma1;
    x-destruir=30;
    x-imagen="/home/usuario/imágenes/arma1.jpg";
    x-tipo=x-arma;
},
{
    x-nombre=rosa1;
    x-imagen="/home/usuario/imágenes/rosa1.jpg";
    x-tipo=x-bonus;
    x-creditos=40;
},

```


</design>

ESQUEMA DEL ARCHIVO DE CONFIGURACION DEL ESCENARIO

Cada etiqueta o atributo puede venir en cualquier orden.

- X es la posición horizontal (Columnas).
- Y es la posición vertical (Filas).
- Ancho es la línea horizontal
- Alto es la línea vertical Ejemplo un tablero de 10x10

[illegible]

ESTRUCTURA

```
<x-escenarios background=Nombre_Fondo; ancho=n; alto=n>
<x-personajes>
  <x-heroes>
    Nombre_heroe ( posicion_x, posicion_y);
  </x-heroes>
  <x-villanos>
    Nombre_villano( posicion_x, posicion_y);
    Nombre_villano ( posicion_x, posicion_y);
  </x-villanos>
</x-personajes>

<x-paredes>
  Nombre_bloque(PosiciónInicial_x..PosiciónFinal_x,posición_y);
  Nombre_bloque(Posición_x,PosiciónInicial_y..PosiciónFinalY);
  Nombre_bloque(Posición_x,Posición_y);
</x-paredes>

<x-extras>
  <x-armas>
    Nombre_arma(posición_x,posición_y);
    Nombre_arma(posición_x,posición_y);
  </x-armas>
  <x-bonus>
    Nombre_bonus(posición_x,posición_y);
  </x-bonus>
</x-extras>

<meta>
  Nombre_meta(posición_x,posición_y);
</meta>
</X-escenario>
```

ETIQUETA DE PERSONAJES

Únicamente se podrá seleccionar un héroe o personaje principal, este podrá ser manipulado por el usuario que proba la aplicación.

Se podrán seleccionar n-enemigos, que al ejecutar la aplicación tendrán movimiento automático.

<x-Personajes >

<x-heroes>

Rey(4,8);

</x-heroes>

<x-villanos>

soldado(9,10);

soldado(3,3);

soldado(14,3);

</x-villanos>

</x-Personajes >

ETIQUETA DE PAREDES

Proporcionará una pared para crear un laberinto para darle mayor complejidad al jugador de llegar a la meta. Estarán limitadas por el tamaño del escenario nombre(x,y).

Estas pueden ser indicadas ya sea por un punto (x,y) o punto inicial x hacia un punto final x o su equivalente en y esto con el fin de construir cuadros continuos más fácilmente que se desplazan horizontalmente pero verticalmente siguen igual.

<paredes>

suelo1(0,11);

suelo1(1..2,11);

suelo1(0..2,10);

suelo1(4,9..11);

suelo1(5,10..11);

suelo1(7..12,11);

suelo1(12,11..10);

</paredes>

ETIQUETA DE EXTRAS

Estas pueden ser ya sea objetos de bonus o de destrucción que fue descrito anteriormente. se pueden agregar n-objetos en cualquier posición.

<extras>

<armas>

arma1(10,10);

bomba1(1,9);

</armas>

<bonus>

Diamante1(8,5);

Diamante1(5,3);

</bonus>

</extras>

ETIQUETA DE META

El personaje al llegar y tocar la meta habrá terminado y ganado el juego.

<meta>

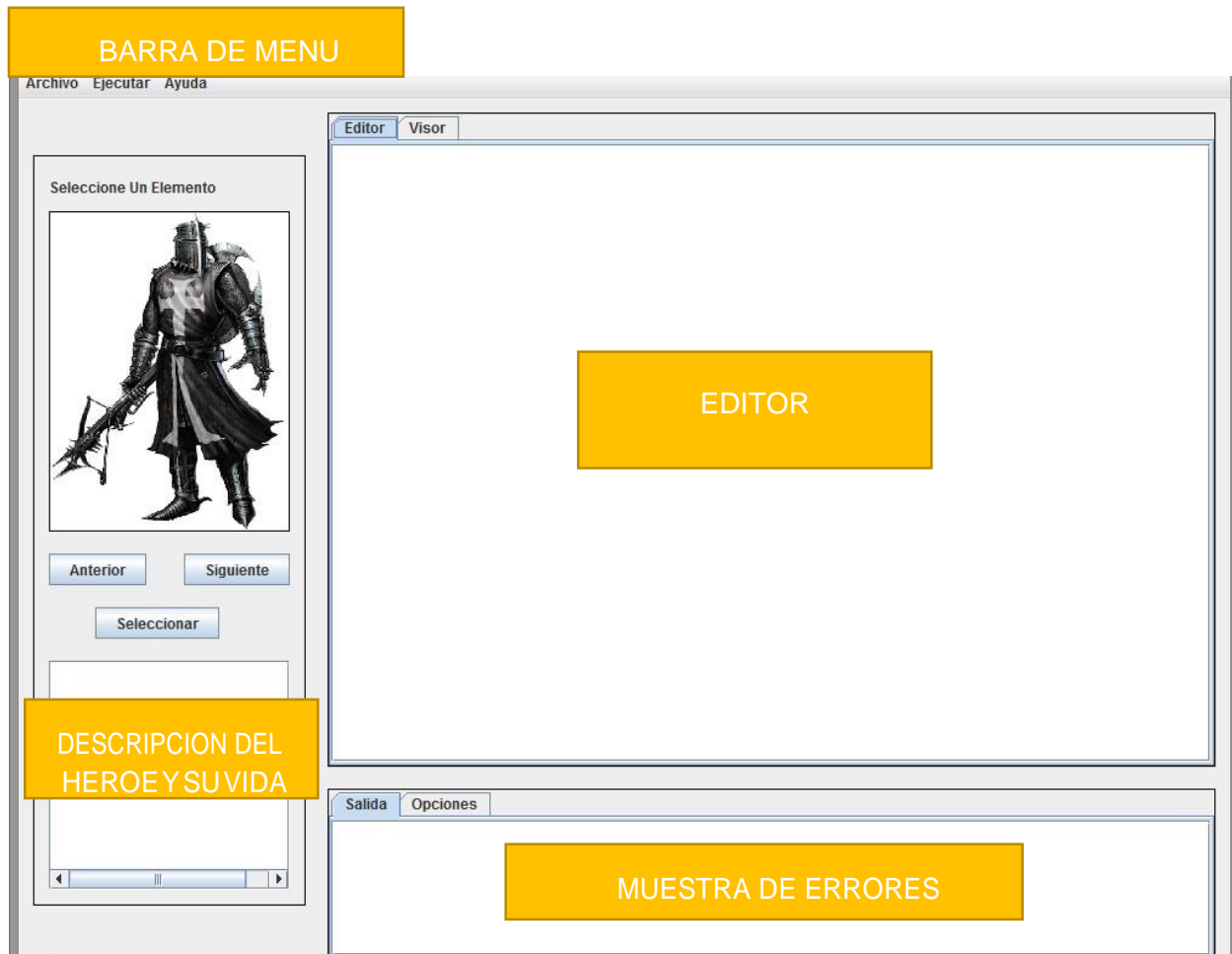
salida1(0,2);

<meta>

APLICACIÓN

INTERFAZ GRAFICA

Diseñar una interfaz amigable y cómoda que facilite la creación de escenarios, estos escenarios posteriormente de ser creados se podrá jugar.



Probar y Ejecutar

Los movimientos del personaje principal o héroe como mínimo deberán poder realizar:

Estas son las teclas mínimas que se deberán usar, pero se pueden agregar más teclas para facilitar su funcionamiento.	Flecha Izquierda, Flecha Derecha, Flecha Arriba, Flecha de abajo, Letra a: disparo.
---	---

Villanos

Se moverán automáticamente, en un rango de casillas y volverán a su posición inicial.

Si implementa un algoritmo que mejore la IA de los enemigos y estos puedan desplazarse en busca del héroe y que tengan un movimiento que no sea cíclico, se dará un **BONUS EXTRA** sobre los 100 puntos de la nota del proyecto.

Componentes informativos de la pantalla de juego.

Como mínimo debe contar con los siguientes componentes gráficos que informen al jugador del progreso de la partida.

- Barra de vida del héroe.
- Numero de enemigos destruidos.
- Nombre del héroe.

BARRA DE MENU

La aplicación deberá de mostrar como mínimo los siguientes menús. Tomando en cuenta que deberá de ser totalmente gráfica.

Archivo	Ejecutar	Ayuda
Nuevo	Compilar archivo de configuración	Manual de Usuario
Abrir	Compilar archivo de carga de escenario	Manual Técnico
Guardar	Ejecutar juego	Acerca de
Guardar Como	Errores	
Salir	Tabla de Símbolos	

Las especificaciones del menú serán de la siguiente forma

- Archivo
 - Nuevo: Despliega un nuevo documento en blanco para editar.
 - Abrir: Permite abrir un programa con las extensiones especificadas.
 - Guardar: Permite guardar el programa editado actualmente con la extensiones especificadas.
 - Guardar Como: permite guardar el programa actual con otro nombre.
 - Salir: con esta opción cerramos la aplicación.
- Ejecutar
 - Compilar archivo de configuración: Compila el archivo de configuración que se encuentre abierto en el editor.
 - Compilar archivo de carga de escenario: Compila el archivo de configuración de escenarios que se encuentra abierto en el editor.

- Ejecutar: Al momento de compilar los dos archivos se podrá ejecutar el juego.
- Errores: Mostrara los errores léxicos, sintácticos y semánticos.
- Tabla de símbolos: Muestra todos los personajes y objetos creados bajo el archivo de configuración.
- Ayuda
 - Manual de usuario: Se deberá mostrar el manual de usuario, para guiar a la persona en la utilización de la aplicación.
 - Manual técnico: Se deberá mostrar el manual técnico, para entender como se ha realizado la aplicación.
 - Acerca de: Permite mostrar información de la persona que ha creado la aplicación.

CONSIDERACIONES

- El proyecto debe ser realizado con el lenguaje C# con la herramienta **Irony** en caso contrario no tendrá derecho a calificación.
- Los archivos tendrán extensión *.xconf(configuraciones) y *.xesc (escenarios) respectivamente, para que la aplicación los reconozca.
- El lenguaje de configuración y escenarios es No case-sensitive.
- La realización del proyecto es individual, **está prohibido que los estudiantes trabajen en grupo y se compartan el proyecto**, copias de proyecto tendrán un valor de 0 (cero) y serán reportados al ingeniero y a la escuela de Sistemas.
- El archivo a entregar debe contener lo siguiente:
 - El proyecto de completo de Visual Estudio
 - El nombre de la aplicación debe contener el siguiente formato Proyecto1_<numero_carnet>
 - Archivo ejecutable, probado dado que no se calificara a través de ningún IDE de desarrollo.
 - Archivo de Entrada, que sean reconocibles por su aplicación.
 - Manuales de Usuario y Técnico.

La entrega será el día 26/12/2018, el horario de entrega y de calificación se estará informando por la plataforma Classroom.