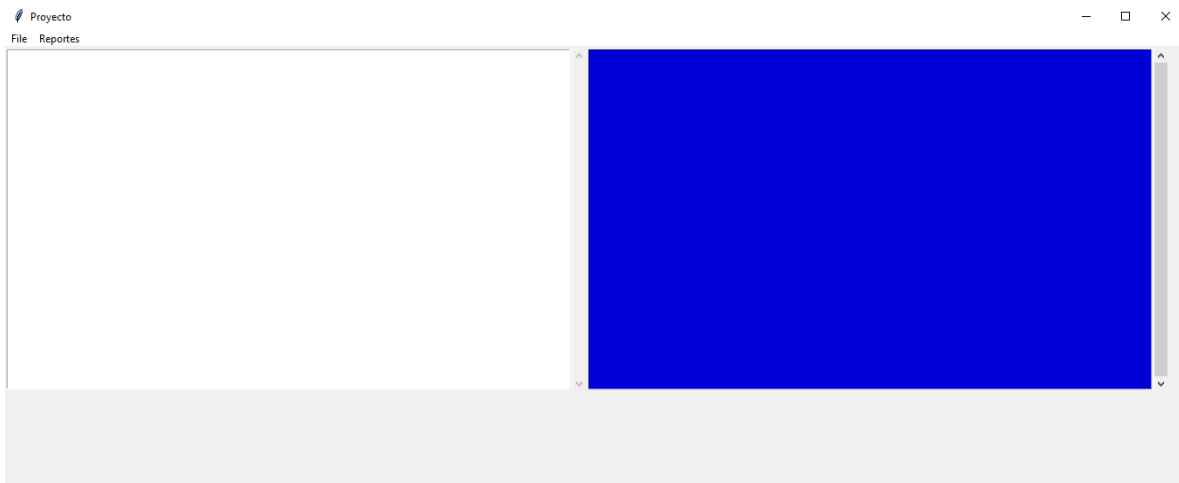


## Manual de Usuario

### Descripción:

El lenguaje JPR debe poder realizar instrucciones de alto nivel las cuales están definidas en el nuevo lenguaje que será exclusivo para la USAC, este se utilizará para los alumnos que estén cursando el primer curso de programación, sirve para que se puedan ver las funcionalidades principales y la aplicación de los fundamentos de programación.

### Funcionamiento de la aplicación:



La aplicación cuenta con dos cuadros de texto, los cuales se verán de distintos colores, el blanco será el editor. En este se puede agregar código permitido por el lenguaje para poder realizar una traducción. En el de color azul se verán los resultados de dicho texto escrito en el editor. También contará con dos diferentes menús que servirán para poder realizar un análisis o crear nuevos archivos, guardarlos, abrirlos etc. Y otro que será utilizado para mostrar los diferentes reportes como son el Reporte de errores, Reporte de un AST de la aplicación y un Reporte de la Tabla de Símbolos creada.

```
var a = 0
main() {
  a = 3
  print("valor de a: " + a)

  var b = a + 4
  print("valor de b: " + b + " ;D")
}
```

```
valor de a: 3
valor de b: 7 ;D
```

Se pueden Realizar diferentes operaciones aritméticas como por ejemplo sumas, restas, multiplicación, división, potencias y módulos.

Además de poder asignarle esos valores a variables creadas con anterioridad.

```

var a = True
main(){
print("valor de a: " + a)

var b = a && False
print("valor de b: " + b + " D:")
}

```

```

^
valor de a: True
valor de b: False D:

```

También esta permitido realizar operaciones lógicas como lo son and, or y not las cuales pueden tener valores de True o False, son útiles para la realización de if's, while's, for's, etc.

```

var a = 7
main(){
if(a < 10){
print("El menor es a con un valor de: " + a + " D:")
}else{
print("El mayor es a con un valor de: " + a + " D:")
}
}

```

```

^
El menor es a con un valor de: 7 D:

```

También están las operaciones Relacionales, las cuales pueden ser útiles para comparar valores junto las funciones if que tienen también un apartado de else y else if.

```

main(){
var r = 0
while(r < 2){
for(var a = 0;a<2;a++){
print("valor de a: " + a)
}
r++;
}
}

```

```

^
valor de a: 0
valor de a: 1
valor de a: 0
valor de a: 1

```

Contiene ciclos, los cuales son while y for, estos deben ir acompañados de una función de incremento ya que si no se utiliza podrían quedarse las funciones en un bucle y saturar la memoria RAM.

```

main(){
print(suma(3,5))
}
func suma(int a,int b){
return a+b
}

```

```

^
8

```

Cuenta con funciones las cuales pueden ser llamadas dentro de un método main o dentro de cualquier otra función, estas pueden tener parámetros y pueden ser utilizados dentro de ellas a su conveniencia o criterio del programador.

```

main(){
int [] a = new int [2]
a[0] = 8
print(a[0])
}

```

```

^
8

```

Se pueden declarar arreglos y se puede tener un acceso a ellos para poder modificar y consultar los datos ingresados en cierta posición de este.

```
main(){  
int [] a = new int [2]  
a[0] = 5  
print((string)a[0])  
}
```

5

Se pueden realizar casteos de ciertos tipos.

```
main(){  
int [] a = new int [2]  
a[0] = 5  
print(length(a))  
}
```

2

Se puede obtener el valor del tamaño de cadenas y arreglos así como también existen otras funciones nativas como por ejemplo Round que es para redondear números o TYPEOF para obtener el tipo de la variable.