**qwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnm**

|  |
| --- |
| V#  Lenguaje de programación  Víctor Gil Ruiz |

Índice

Contenido

[Descripción del sistema 3](#_Toc137944601)

[Identificación de requisitos del sistema 3](#_Toc137944602)

[Requisitos de información 3](#_Toc137944603)

[Requisitos funcionales 4](#_Toc137944604)

[Descripción de la solución 5](#_Toc137944605)

[Manual 6](#_Toc137944606)

[ESTUDIO DEL COSTE DEL PROYECTO 7](#_Toc137944607)

Introducción

V# consiste en un lenguaje de programación compilado, el cual está basado en Python y C.



Este está diseñado para programadores que vienen de otros idiomas, ya que se asemeja a los más comunes (C, Java,…); o para las personas que lo están aprendiendo, ya que, mezcla y reúne lo mejor de varios lenguajes, ya sea como la facilidad, portabilidad y lo liviano de Python y la sintaxis de C o java (común en varios lenguajes). Esto hace que a la hora de adaptarse a otros lenguajes sea más fácil, rápido y menos pesado.

Estudio de viabilidad

# Descripción del sistema

El proyecto de V# se compone de dos grandes pilares, como son el IDE, y el script. El IDE, diseñado en Delphy, permite lanzar, crear y editar texto. El script traduce, avisa de errores y lanza en Python el resultado, por consola.

# Identificación de requisitos del sistema

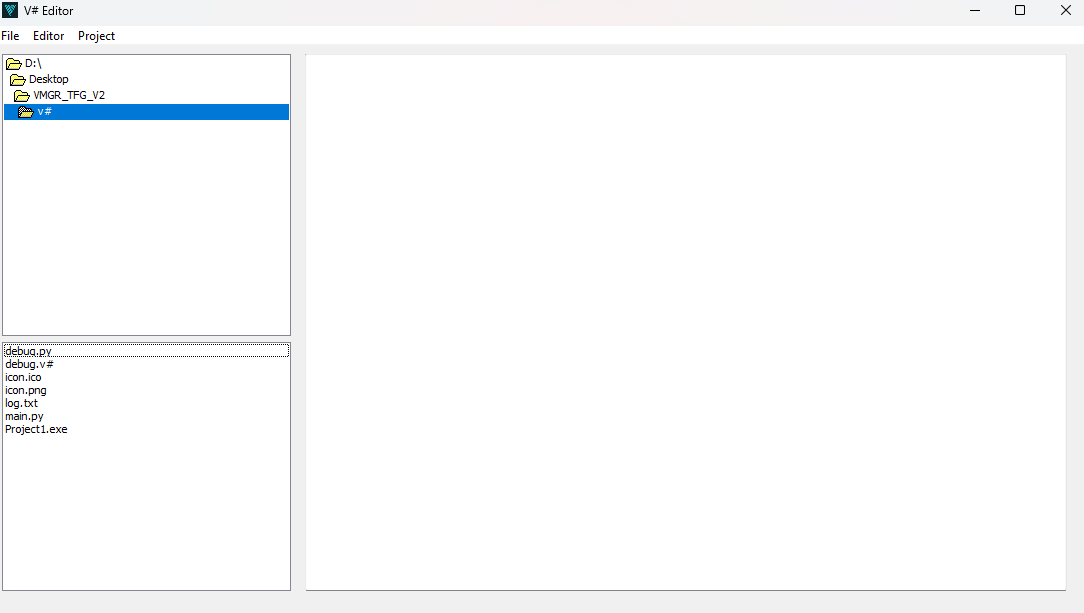
## Requisitos de información

V#, como se mencionó anteriormente, trabaja con Delphy y Python.

* En Delphy se usa distintos objetos, como son TRichtext, para el editor, TMenu para el menú y sus objetos, el cual permite hacer acciones varias, como son guardar, ejecutar, cargar, crear,…, y administradores de directorios y archivos, el cual, están unidos por el enlace. Tambien lanza comandos de CMD para actuar con el script
* En Python, el script tiene la mayor carga del peso. Este se encarga de traducir, mostrar errores y lanzar el resultado. Este script esta diseñado de manera modular, es decir, que si se quiere añadir reglas posteriormente, las cuales se van a crear, tiene fácil implementación

## Requisitos funcionales

En cuanto a IDE, la funcionalidad es sencilla, la mayoría de las gestiones se realizan desde el menú, también se puede cambiar de archivos desde el gestor de archivos ubicado a la izquierda, el cual te desglosa en carpetas y archivos de la ruta que especialices de forma grafica



En la parte derecha, Esta el editor de texto, el cual permite hacer gestiones de texto tales como copiar, pegar, deshacer y rehacer.

En el menú, puedes lanzar desde la opción *¨project¨*, también puedes cargar el ultimo que se ha ejecutado, esto se usa para recuperar el trabajo después de un cierre inesperado.

# Descripción de la solución

El uso de librerías es indispensable para cualquier proyecto medianamente grande, y este no es una excepción.

En Delphy se ha usado la librería ¨Shell¨, el cual permite lanzar comandos por consola, esto nos permite comunicarnos con el script, ya sea como ejecutarlo, como pasarle información en modo de parámetro, que luego dentro del script podemos leer y tratarlo.

En Python usamos un par más, entre ellos están:

* SYS: este nos facilita saber los parámetros que usamos en el lanzamiento
* OS: este se usó para conocer la ruta de distintos archivos, los cuales son fácilmente desglosables para crear o leer archivos en la misma ubicación
* SUBPROCESS: Se usó para lanzar comandos en consola, y con ello, lanzar el script de Python resultante

Este sería el esquema del trayecto de la información:

Ejecución

Resultado

Script

Log de fallos

Código V#

# Manual

V#, basado en Python tiene algunas variantes mostradas en la siguiente tabla, con ejemplos en Python y en V#. El funcionamiento solo cambia en los vswitch, ya que en Python solo existe en la versión mas reciente.

|  |  |  |
| --- | --- | --- |
|  | PYTHON | V# |
| if | If condición: | If(condición){  } |
| switch |  | Vswich(condición){  Case condición:  Case condición:  } |
| while | While condición: |  |

# ESTUDIO DEL COSTE DEL PROYECTO

Este proyecto fue realizado durante el periodo de Practias, por lo cual, la relación trabajo/días es muy baja, ya que no era mi única dedicación en ese momento

Igualmente se desglosara el tiempo utilizado en el proyecto:

-IDE: 40 horas

-Script: 45 horas

-investigación: 10 horas

-documentación: 5 horas

-preparación para defensa de proyecto: 8 horas

TOTAL:108

Coste por hora: 15€

TOTAL COSTE= 1620€