

Taller de Programación GR 6

Semestre 01 – 2019

Integrantes:

Sebastián López Herrera

2019053591

Daniel Sequeira Retana

2019039641

Tarea Programada #2

Entrega: martes 14 de mayo de 2019

Estatus: Excelente

Enunciado del proyecto:

Un cliente nos contrata con la esperanza de que le desarrollemos un programa de repositorio de frases célebres de **Star Wars**, debido a que es un fanático de las películas.

El cliente nos solicita que las frases deben de provenir de un API que una amiga diseño, este API se encuentra en la URL <http://swquotesapi.digitaljedi.dk/index.html>. Considere que el TEC, bloquea esta ruta de trabajo.

El cliente posee cierto conocimiento en el tema de la programación informática, por lo que nos solicita las siguientes estructuras del sistema:

Estructura: Matriz

Él sabe que el API de su amiga retorna la información como texto plano, en el cual se incluye un "Id" de la frase, además de la frase deseada y del personaje que la dijo. El cliente quiere una matriz de la siguiente forma:

Nombre del personaje **Lista** de frases **Lista** de los id's Código de aplicación

Esta matriz debe de expandirse y o modificarse cada vez que la llamada a la API se realice, insertando los datos obtenidos. De forma inteligente, el sistema debe saber si la frase retornada del API ya está en el sistema y no permitir duplicados.

Estructura: Diccionario

Además, quiere que el programa posea un diccionario, donde la llave es el nombre del personaje y el contenido es una lista compuesta por:

Código de la aplicación que el cliente especifico como: El símbolo "#" seguido por la primera letra del nombre del personaje, seguido por un número serial de 000 a 999, que debe crecer automáticamente, seguido por un "-", terminando en la última letra del nombre del personaje. Ejemplo, si el quinto personaje encontrado en la aplicación es "Yoda" el código de la aplicación sería: #Y005-A

Número de resultados que el API ha dado, de ese personaje para propósitos estadísticos. Aquí debe aumentarse el contador a pesar de aparecer frases repetidas.

Compartir datos

Por último el cliente quiere poder compartir y guardar las frases que ya obtuvo, por lo que piensa que la mejor forma de realizarlo es mediante archivos extensión "**XML**", por lo que nos dice que hay dos formas de hacerlo:

BackUp:

El usuario al cerrar la aplicación, se le da la opción de hacer un backup, por lo que, si el usuario selecciona que si lo desea, la aplicación crea un archivo "**XML**", llamado "**BackUp**", el cual la próxima vez de que se abra la aplicación, debe leer y cargar en la aplicación. En caso de existir previamente, debe controlar estratégicamente esta solución.

Share:

El usuario selecciona la opción de share, por lo que selecciona las frases a compartir

y crea un archivo “**XML**”, llamado “share-fechaYmomentoDeCreacion.xml”, el cual contiene UNICAMENTE las frases seleccionadas por el usuario. Ejemplo: share-dd-mm-aaaa-hhmm-ss.xml (Día, mes, año, horas, minuto y segundo).

Pantalla Principal:

Se espera que el programa principal posea una pantalla como la del siguiente ejemplo:

La pantalla principal debe:

- (1) Mostrar TODAS las frases que se han buscado, las cuales son acumulativas (se pueden hacer 2 búsquedas de 2 frases cada una y al final se deben de mostrar 4 frases diferentes). Debe cargar de “**BackUp**” si la aplicación ya fue antes usada y el usuario aceptó guardar los datos o aparecer esta sección vacía, pues es la primera vez que se usa la aplicación o no se desearon guardar los datos.
- (2) El cliente, además pide que se puedan hacer llamadas repetitivas a la aplicación, que se puedan realizar desde una petición de una frase hasta 50 frases, las cuales por el tema del diseño de la API se sabe que pueden incluir frases repetidas, pero que se deben de manejar.
- (3) “**Share**” debe dejarme escoger las frases a ingresar en el archivo XML y crearlo (diseño de dicha pantalla o forma de escoger las frases se deja creatividad del grupo) y enviar como adjunto al correo electrónico del correo deseado.
- (4) Debe mostrar la estadística de quien es el personaje con más frases en el sistema, esto se obtiene con los datos registrados en el diccionario, aquí si el API arrojó datos repetidos, allí debe considerarse las repeticiones.
- (5) En el momento de cerrar el programa debe de preguntar si gusta hacer el backup de las frases actuales ubicadas en la sección (1).

Consideraciones importantes

- Debe considerar el manejo de errores. Por ejemplo, en caso de que no exista conexión a internet para consultar los datos o la validación de la dirección de correo electrónico.
- Debe indicar el nombre y dirección URL de las librerías que utilizó a fin de configurar adecuadamente el ambiente de pruebas para su revisión, en caso de que se requiera.
- En caso de que el archivo **BackUp.xml** no exista, debe crearlo en el directorio local.
- Agregue un “**Acerca de...**” para consignar la información de los integrantes del equipo y llamar desde allí el **manual de usuario**.

Scrum

Archivo de Drive

Scrum

Fecha Inicial: 23/04/2019

Grupo de Trabajo: Sebastián López Herrera y Daniel Sequeira Retana

Scrum Máster: Juan Jose Solano

Fecha: 23/04/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Escuchar la presentación de la tarea programada
¿Qué haré a continuación?	Leer el documento para entenderlo
¿En qué ocupo ayuda?	Plantear cómo desarrollar la tarea

Fecha: 24/04/2019	1
Nombre: Daniel Sequeira Retana	
¿Qué he hecho?	Releer las instrucciones para entenderlas mejor
¿Qué haré a continuación?	Investigar a cerca de tkinter
¿En qué ocupo ayuda?	Donde encontrar buen material para las POO en python

Fecha: 24/04/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Investigación sobre tkinter y pygame
¿Qué haré a continuación?	Crear repositorio en GitHub
¿En qué ocupo ayuda?	Desarrollar esquema de la tarea

Fecha: 25/04/2019	2
Nombre: Daniel Sequeira Retana	
¿Qué he hecho?	Buscar información sobre Tkinter
¿Qué haré a continuación?	Practicar ejemplos básicos de dicha librería
¿En qué ocupo ayuda?	Cómo trabajar con tkinter

Fecha: 26/04/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Crear el repositorio en GitHub
¿Qué haré a continuación?	Investigar sobre HTTP
¿En qué ocupo ayuda?	Ensamblar la pagina web con el programa

Fecha: 26/04/2019	3
Nombre: Daniel Sequeira Retana	
¿Qué he hecho	Crear una cuenta en github
¿Qué haré a continuación?	Instruirme con python y conexión a internet
¿En qué ocupo ayuda?	Material para estudiar y practicar este tema

Fecha: 29/04/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Cree funcion para sacar una lista con todas las respuestas de la API
¿Qué haré a continuación?	Investigar sobre algoritmos para verificar si el usuario posee
¿En qué ocupo ayuda?	Ensamblar la pagina web con el programa

Fecha: 29/04/2019	4
Nombre: Daniel Sequeira Retana	
¿Qué he hecho	Lograr enviar correos desde python
¿Qué haré a continuación?	Empezar a trabajar con XML y lectura de correos
¿En qué ocupo ayuda?	Leer correos desde python y Material para XML y sus formas de trabajarlo

Fecha: 1/05/2019	5
Nombre: Daniel Sequeira Retana	
¿Qué he hecho	Aprender XML, impresiones básicas en xml
¿Qué haré a continuación?	Xml avanzado + escritura
¿En qué ocupo ayuda?	Nodos y atributos de xml

Fecha: 02/05/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Crear la matriz y el diccionario
¿Qué haré a continuación?	Conectar todo a una interfaz gráfica
¿En qué ocupo ayuda?	Aprender más sobre tkinter

Fecha: 03/05/2019	6
Nombre: Daniel Sequeira R	
¿Qué he hecho?	Aprender a leer nodos de xml
¿Qué haré a continuación?	Aprender a escribir nodos con xml
¿En qué ocupo ayuda?	Escritura de archivos xml

Fecha: 04/05/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Rediseñar la matriz y diccionario
¿Qué haré a continuación?	Conectar todo a una interfaz gráfica
¿En qué ocupo ayuda?	Aprender más sobre tkinter

Fecha: 5/05/2019	7
Nombre: Daniel Sequeira Retana	
¿Qué he hecho?	Escribir y leer en el xml, enviar por correo todo el xml
¿Qué haré a continuación?	Leer y cargar un back up desde el correo
¿En qué ocupo ayuda?	Lectura de correos en python

Fecha: 06/05/2019	8
Nombre: Daniel Sequeira R	
¿Qué he hecho?	Lectura de correos en python
¿Qué haré a continuación?	Mejorar la impresion
¿En qué ocupo ayuda?	Como evitar la conversiona byte que a;ade un b'

Fecha: 08/05/2019	9
Nombre: Daniel Sequeira R	
¿Qué he hecho?	Mejora en las funcines en general
¿Qué haré a continuación?	Mejorar la escritura del xml para una mayor compresion
¿En qué ocupo ayuda?	Editar xml como parametro sin afectar su integridad

Fecha: 08/05/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Darme cuenta que todo lo hecho es inservible
¿Qué haré a continuación?	Palmarla hasta tener algo que si sirva
¿En qué ocupo ayuda?	No dejar morir el curso

Fecha: 08/05/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Crear las funciones y variables necesarias para el uso del programa
¿Qué haré a continuación?	Conectar la GUI con el programa
¿En qué ocupo ayuda?	Sigo sin poder imprimir un texto en un Text()

Fecha: 11/05/2019	
Nombre: Sebastián López Herrera	
¿Qué he hecho?	Terminar la Interfaz Grafica
¿Qué haré a continuación?	Conectarla a las funciones
¿En qué ocupo ayuda?	Imprimir texto en un Text()

Estadísticas de tiempo

Sebastián López Hererra

Actividad Realizada	Horas
Análisis de Requerimientos	3 h
Diseño de Algoritmos	1 d
Investigación de Tkinter y HTTP Request	3 d
Programación	8 d
Documentación Interna	3 h
Pruebas	7 h
Elaboración del manual de Usuario	5 h
Elaboración de la documentación del Proyecto	2 h
Total	1 semana, 5 días, 19 horas y 59 minutos

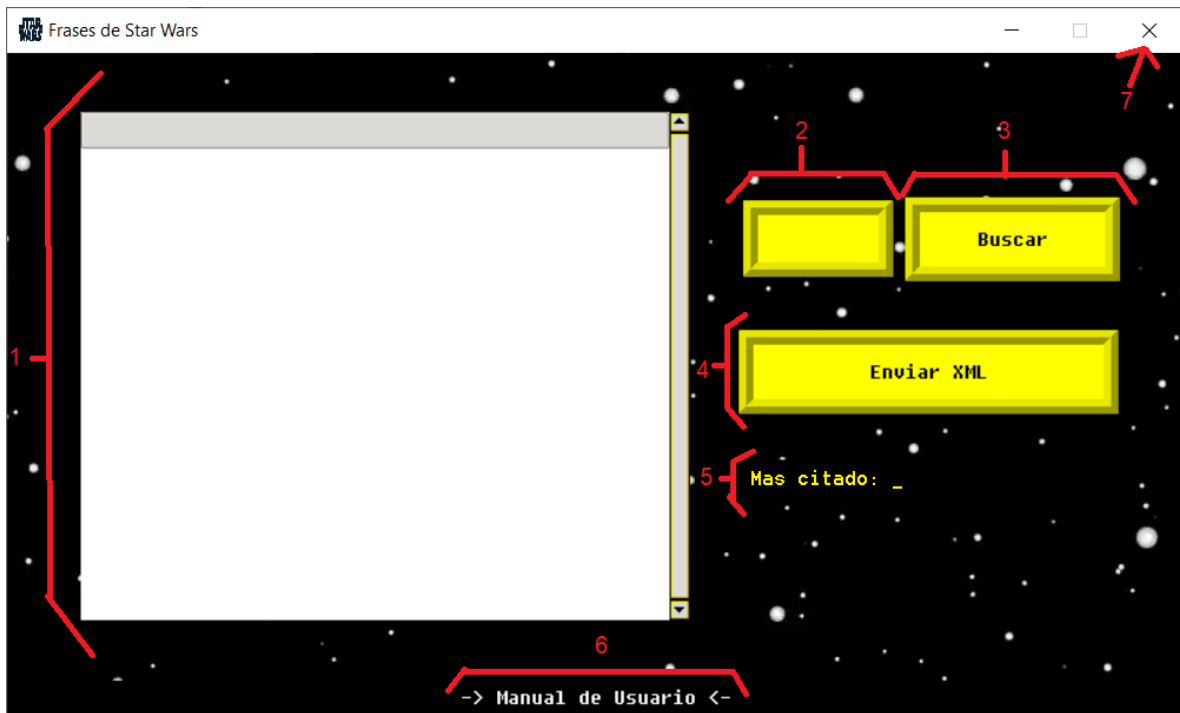
Actividad Realizada	Horas
<i>Análisis de Requerimientos</i>	2 h
<i>Diseño de Algoritmos</i>	15 h
<i>Investigación de XML y tkinter</i>	2 d
<i>Programación</i>	5 d
<i>Documentación Interna</i>	3 h
<i>Pruebas</i>	6 h
<i>Elaboración del manual de Usuario</i>	4 h
<i>Elaboración de la documentación del Proyecto</i>	5 h
Total	1 semana, 1 día, 10 horas y 57 minutos

Manual de Usuario

El programa adjuntado a este manual permite al usuario sacar un numero definido de frases de una API que brinda frases celebres de personajes de *Star Wars*, enviarlas por correo electrónico y hacer una copia de seguridad de estas frases. A continuación, se explicará cómo acceder a cada una de las funcionalidades del programa y el respectivo uso adecuado para cada una.

Cabe resaltar que el correcto funcionamiento de este software requiere de una conexión de Internet estable, ya que requiere de la misma para realizar todas sus funcionalidades. Por lo que es indispensable que se certifique de estar conectado a Internet al momento de iniciar el programa.

Primeramente, al ejecutar el programa, se desplegará una ventana similar a esta:



Como se nota en la imagen anterior hay 7 elementos interactivos en el programa de los cuales va a poder dar uso siguiendo las especificaciones, estos elementos son:

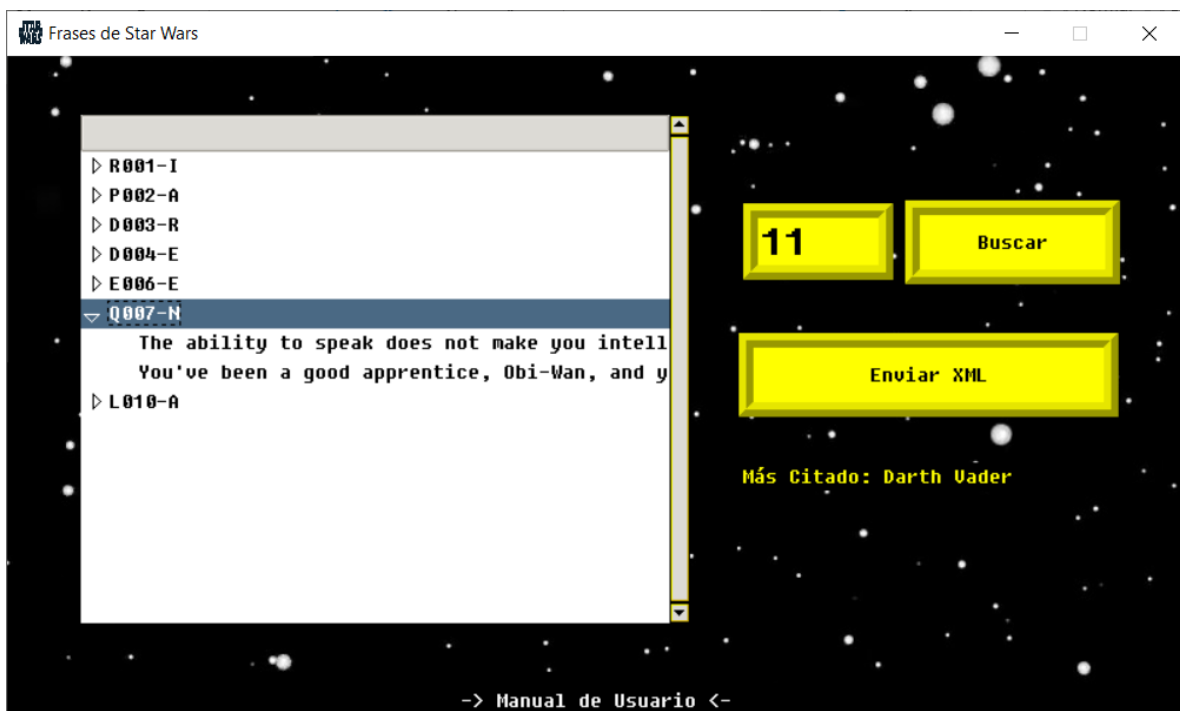
1. Visualizador en cascada
2. Cuadro de texto – Buscar
3. Botón Buscar
4. Botón Enviar XML
5. Cuadro de texto – Diccionario
6. Botón Manual de Usuario
7. Botón Salida

Buscar frases:

Si desea buscar frases, primero deberá posicionar su cursor sobre la caja de texto “Cuadro de texto- Buscar” (2) y darle click izquierdo. A continuación; debe introducir un número, positivo y sin letras y darle click izquierdo al botón a su derecha, “Botón Buscar” (3), el proceso requiere de un tiempo para realizarse, cuando este termine vera a continuación algo parecido:



Para poder ver las frases que recolecto el programa deberá dar en el triángulo que se encuentra a la par de cada código en (1), en este caso, si usted deseara ver las frases asignadas a Q007-N, vera lo siguiente al darle al triángulo:



Como se logra apreciar, se desplegaron 2 frases que eran las que estaban asociadas a Q007-N. No debe cerrar esa carpeta para ver las otras, puede abrir las que desea visualizar.

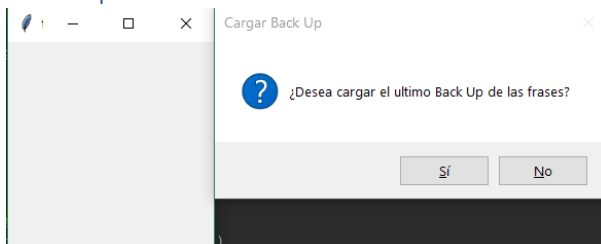


Ya que el texto dentro de (1) supera la cantidad de líneas que puede mostrar, se activa el *scroll* que permite navegar a lo largo de la ventana (1) usando la rueda del mouse o tocando directamente las flechas de la barra.

Desplegar Manual de Usuario

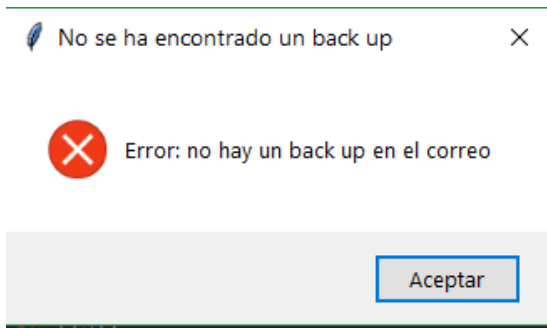
En la parte de debajo de la ventana se encuentra un botón (6) que al ser presionado con el click izquierdo sobre el texto se abrirá inmediatamente el pdf con el manual de usuario.

Back up de inicio:



Al ejecutar el programa nos preguntará si deseamos cargar in back up del correo, si tenemos uno ya enviado al correo este cargará, de lo contrario indicará que no hay mediante un cuadro de texto. Si presionamos Si cargara el XML con las frases previamente enviadas,

si presionamos no simplemente saltara al menú principal

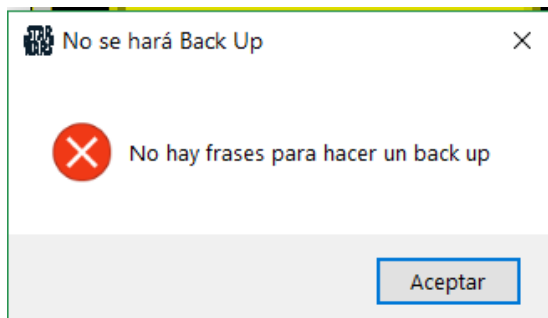


Este será el mensaje indicando que no hay un back up disponible, procederemos a Aceptar y saltaremos a la ventana principal

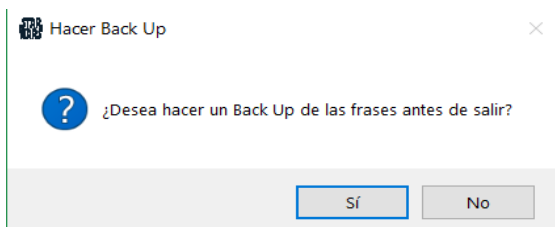
Boton Share:



Al presionar el botón Share enviara a nuestro correo el xml de las frases formadas



En caso de que no hayan frases en el el treeview no dejará hacer el Share



Back up al salir: si presionamos a la X de salir no tirará este mensaje preguntándonos si queremos hacer un back up de las frases que tenemos

Si presionamos a No inmediatamente cerrará la ventana



May the force be with you



Se ha enviado un Back up a tu correo

Aceptar

Si presionamos si nos arrojará un mensaje de confirmación



No se hará Back Up



No hay frases para hacer un back up

Aceptar

En caso de que no hayan frases entonces dira que no puede hacer el back up