

Proyecto final - INFORMÁTICA II

Objetivo

Evidenciar las habilidades adquiridas por el estudiante, en el manejo, manipulación y capacidad investigativa, de las diferentes librerías estudiadas, a lo largo del semestre, y su implementación con el paradigma de programación POO.

Descripción del trabajo a realizar

Crear un aplicativo de escritorio que procese y manipule archivos **mat, csv, dcm, jpg y png**. El aplicativo debe tener nombre, que uds se inventarán (sean creativos), también deben personalizarlo y mostrarán al ejecutarlo. Debe contar con una interfaz gráfica que:

1. Solo pueda ingresar un usuario que este previamente ingresado (dicha información se debe validar desde un Usuario.json que contenga integrantes con una contraseña asociada).
2. Tenga un menú inicial que dirija según el tipo de archivo a una interfaz diferente. Esta se abrirá tras logearse previamente.
 - a. Archivos DICOM
 - b. Archivos JPG, PNG
 - c. Archivos CSV
 - d. Archivos MAT
 - e. Salir (confirmar)
3. Las funcionalidades asociadas a los diferentes botones del menú deben:
 - a. Para el caso de los archivos DICOM, debe contar con una interfaz que visualice las imágenes en los 3 ejes; Sagital, coronal y axial, independiente del archivo que se cargue (En el interior de la interfaz no se aceptan ventanas emergente creadas con `plt.show()`), estas imágenes se deben poder visualizar usando la herramienta slider (como lo explica el monitor en el taller). Y debe mostrar y guardar la información del paciente (solo la del paciente como se hizo en uno de los quices), en una base de datos de su preferencia (MYSQL o MONGO). Esta la deben incluir en la entrega
 - b. Para el caso de la imágenes JPG y PNG, la interfaz debe poder permitir hacer procesos de erosión, dilatación, cierre y apertura donde el usuario pueda manipular

el tamaño de kernel , adicional deben INVESTIGAR sobre algún otro método o librería que no se haya mostrado en clase e implementarlo en esta sección de imágenes en este formato. **NOTA:** Si implementan una librería diferente pueden adaptarla al aplicativo en general como uds la consideren.

- c. Para el caso de los archivos MAT debe permitir cargar cualquier archivo .mat y debe mostrar las llaves asociadas en un QComboBox, para que el usuario elija la llave correspondiente al array, en caso equivocarse que saque un QMessageBox con un mensaje que diga no es un arreglo, vuelva a intentarlo. En caso de elegir la llave correcta , debe visualizar la señal en la ventana no por fuera de esta , adicional la interfaz debe poder permitir graficar un canal o segmento de canales en un intervalo dado por el usuario. En el grafico se debe mostrar el título del grafico, las leyendas y los nombres de los ejes.
 - d. Finalmente para el archivo CSV debe permitir cargar un archivo csv, hacer 2 tipos de operación (el que uds quieran) sobre el dataframe, muestra los resultados , debe mostrar la información del dataframe y debe dentro de la interfaz mostrar algun tipo de grafico (boxplot, dispersión, barras, stem, etc.) (**SOLO UNO EL QUE UDS ELIJAN**)
 - e. Para la opción de salir, debe mostrar un QMessageBox con un mensaje de confirmación de salida.
4. Las interfaces, además del nombre ideado por uds, deben tener un diseño personalizado (colores, fuentes letras, iconos, etc..).
5. Finalmente, hacer un video sustentado el proyecto, como si estuvieran vendiendo un producto , que sería su programa, mostrando su funcionamiento:
- a. El video no debe durar más de 18 minutos
 - b. La idea es que expliquen cada parte del código, las clases que implementaron, su lógica, cómo la pensaron y lo hicieron. Lo deben exponer entre todos los miembros del grupo.
 - c. El video lo pueden grabar con Microsof Teams, este les permite subir fácilmente el video y ya queda en la nube y solo sería compartir el link.

Entregable:

1. Diagrama de clases (con LUCIDCHART)
2. Diseño de las interfaces (Son pantallazos de las interfaces que hicieron)
3. Código funcionando (Un comprimido con todos los archivos)
4. Manual de usuario
5. Sustentación (video)

Observaciones

1. La nota se asignará de acuerdo con la calidad de los trabajos realizados verificando las condiciones y comparando entre los diferentes trabajos de manera que se favorezca una sana competencia.
2. Los trabajos que adicionen funcionalidades extras al aplicativo (y las sustenten) a las ya solicitadas, sumaran puntos adicionales para mejorar notas.
3. El proyecto será en los mismos grupos de los seguimientos, (los que estén solos, pueden unirse a un grupo de dos integrantes) **solo aceptaré grupos de máximo 3 personas.**
4. Se debe utilizar la arquitectura MVC, sin esto no se califica.
5. Es claro que NO deben pasárselo entre uds, cualquier intento de fraude será un cero (0.0).
6. El trabajo debe ser cargado en el TEAMS, solo una vez, es decir, uno de los dos integrantes hará la carga a más tardar el día Lunes 27 DE NOVIEMBRE DE 2023 A LAS 11:59PM. IMPORTANTE: EL TRABAJO NO SE RECIBIRÁ DESPUÉS DE ESTA HORA, ES DECIR A LAS 12:00AM ESTE 20% SERÁ CERO (0.0).
7. Si al ejecutar el script hay algún error, sin importar cual sea, el trabajo se calificará sobre 3.7.

UdeA