PROGRAMA DE BIOINGENIERÍA

Bioseñales y Sistemas

Proyecto 1: Sistema básico para la visualización de señales usando la arquitectura MVC

**Objetivos:**

* Aplicar la arquitectura MVC para el desarrollo de proyectos de software
* Implementar algoritmos de análisis espectral
* Trabajar el proyecto apoyado por una herramienta de gestión de código

**Se solicita:**

1. Realizar un minitutorial de gestión de repositorios de código y buenas prácticas de programación (20%)
2. Abrir una cuenta en un repositorio de gestión de código (tipo GIT) y crear un proyecto para gestionar el código de la tarea (20%)

**Además. construir una interfaz gráfica que permita**

1. Cargar una señal con formato .mat
2. Visualizar la señal cargada
3. Analizar la señal carga usando el método Welch o el método Multitaper. La interfaz debe permitir configurar los parámetros de cada algoritmo de estimación espectral tal cual como se indicará durante las clases (20%)
4. Obtener la representación por transformada de Wavelet Continua de la señal, graficarla y desplazarse sobre la misma (20%)
5. Hacer un informe que contenga el análisis en el dominio de la frecuencia de la señal entregada. Este análisis debe contemplar variaciones de los parámetros de los algoritmos de estimación espectral y de las condiciones entregadas de la señal (20%). El informe también debe tener una descripción de cómo se realizó la gestión del código en el repositorio

**Condiciones de entrega:**

* La tarea se debe entregar en parejas
* Tarea sin funcionar no se revisa
* Código sin documentar no se revisa
* Se deben entregar todos los elementos para proceder con la calificación, aunque solo el programa funcionando es calificable.
* Cada sección de código mal diseñado (redundante, variables innecesarias, ciclos sin sentido), rebaja media unidad
* Trabajo sin sustentar no se califica. En la sustentación se debe hacer énfasis en aquellas operaciones que hicieron uso de Numpy
* Cada error en el programa rebaja una unidad.
* Se deben tener más de una wavelet madre como opción en el filtrado
* El programa no debe detenerse si el usuario ingresa mal una opción, debe permitir que vuelva e ingrese la opción hasta que lo haga de manera correcta.
* No deben haber interfaces idénticas o sospechosamente parecidas