

Decisiones tomadas en implementación de la GUI

1. MVC

1.1. Interfaces

Para esta implementación del patrón de diseño MVC se ha modificado ligeramente la idea detrás de las interfaces. En primer lugar el ModeloCanciones solo implementa una interfaz, InterrogaModelo, dicha interfaz declara los métodos que necesita el controlador para obtener datos para la Vista.

La clase ControladorCanciones cambia ligeramente su comportamiento, en esta aplicación el Modelo es creado desde el principio y nunca requiere de cambios. Por esto, el Controlador se transforma en un intermediario en el intercambio de datos entre la Vista y el Modelo. La interfaz Controlador solo declara métodos para que la vista realice preguntas sobre títulos de canciones.

Respecto a la Vista se ha decidido separarla en dos ventanas, cada una de ellas implementa dos interfaces diferentes. La VistaCanciones simplemente es una escena que recoge los datos acerca del algoritmo, distancia y canción que desea el usuario. A continuación crea una instancia de VistaResultado y le pasa como parámetros los datos obtenidos más una referencia de sí misma. Es por esto y el cambio del comportamiento del Controlador que la interfaz InformaVista (implementada por VistaCanciones) no declara ningún método. Finalmente la VistaResultado implementa la interfaz InterrogaVista que declara todos los métodos necesarios por el Controlador para obtener los datos introducidos por el usuario.

2. Aspecto Vista

En primer lugar se decide dividir la Vista en dos escenas. Una para listar las canciones y otra para mostrar las recomendaciones relacionadas sobre la canción seleccionada.

La primera Vista (VistaCanciones) cuenta con tres apartados. El primero permite elegir el algoritmo de ML para buscar las canciones recomendadas. Este grupo de componentes consiste en un Label y dos radioButton excluyentes. El segundo grupo es exactamente igual al primer apartado pero cambiando el significado de los botones, ahora selecciona la distancia empleada en los algoritmos de ML. El tercer apartado consiste de un Label y una ListView de selección única con los títulos de todas las canciones. Al final aparece un botón para recomendar, dicho botón se inicia deshabilitado y únicamente se habilita cuando algún escuchador de los ToggleGroup o el de la lista detecta que se ha seleccionado tanto un algoritmo, una

distancia y una canción. Una vez se clic el botón de recomendar (o se hace doble click en un título de una canción con la distancia y algoritmo seleccionados). se pasa a la segunda Vista (VistaResultado) que contiene un Spinner de valor inicial 5 que cuando es manipulado por el usuario actualiza la ListView de canciones recomendadas que aparecen justo debajo. Al final aparece un botón “Close” para volver a la escena de seleccionar una canción.

Como medidas para facilitar la vida al usuario se ha añadido:

- Un ToolTip a la ListView de canciones para anunciar la funcionalidad del doble-clic.
- La propia funcionalidad del doble-clic.
- El botón de recomendación incluye dinámicamente el nombre de la canción seleccionada.
- El botón de recomendación permanece deshabilitado hasta que se selecciona el algoritmo, distancia y canción.
- El spinner se puede editar por teclado, y solo acepta valores numéricos.
- El spinner tiene un límite de 100 canciones recomendadas por qué consideramos que más de 100 son demasiadas recomendaciones para el usuario promedio.

En conclusión, nuestra aplicación se ha construido en base a dos principios fundamentales. La interfaz gráfica ha de responder inmediatamente (por eso el Modelo guarda un recomendador para cada combinación de algoritmo y distancia) y que no sea necesario probar mucho para entender su funcionamiento, es decir, evitar errores humanos del usuario como no elegir una distancia, no escoger un valor en el spinner, etc.