|  |  |  |
| --- | --- | --- |
| **Objetivos**   1. Generar la habilidad para la algorítmica para localizar elementos en una estructura enlazada sencilla | **Número de participantes**  participante.jpgAntony Salcedo | **Materiales**   * Internet * Eclipse |
| **Duración de la actividad**  **01:30** |
| **INSTRUCCIONES**   1. Importa en eclipse el proyecto de karaoke que acompaña a este laboratorio, ejecuta la aplicación y verifica que todas las opciones funcionan correctamente. **No olvides activar los assert en Eclipse.** 2. **[0.1]** Estudia el modelo conceptual del mundo de la aplicación, el archivo descripción.pdf y responde las siguientes preguntas:  |  |  | | --- | --- | | Pregunta | Respuesta | | ¿Los artistas son modelados mediante qué tipo de lista enlazada? | Bidereccional – Doblemente enlazada. | | ¿Cuántas canciones pueden registrarse a un artista? | No está definido un límite de ingreso para las canciones, de igual forma se está trabajando con referencias, por lo que es una lista que seguirá aumentando hasta que la última canción sea null. | | ¿Cómo se llama la relación que le permite a un artista conocer a su anterior? | La relación que le permite a un artista conocer su anterior artista es **anterior.** | | ¿De qué tipo es el anterior atributo? | El **anterior** atributo es de tipo **Artista.** | | ¿Las canciones de cada artista son modeladas mediante qué tipo de lista enlazada? | Simplemente enlazada o uniderireccional | | ¿Qué relación permite al karaoke acceder a los artistas? | La relación que permite al karaoke acceder a los artistas es **primerArtista.** | | Escriba las categorías musicales que maneja la aplicación del karaoke | 1. Rock 2. Pop 3. Reggae 4. Tropical 5. Electronica | | Escriba los datos requeridos para crear un artista | Los datos requeridos con la categoría, Nombre, Imagen. | | Escriba los datos requeridos para crear una canción | Los datos requeridos para crear una cacncion son: Nombre, Duración (en segundos), Letra, Dificultad (valor entre 1 y 10, donde 10 representa la canción más difícil de interpretar), Ruta (Un archivo con la pista de la canción en formato midi). | | Escriba dos posibles situaciones anormales o de excepción que pueden presentarse en la aplicación | 1. No pueden existir dos artistas con el mismo nombre 2. Un artista no puede tener dos canciones con el mismo nombre | | Escriba todos los servicios que ofrece la aplicación del karaoke | 1. Agregar un artista al karaoke,  2. Agregar una canción a un artista del karaoke,  3. Eliminar un artista del karaoke.  4. Eliminar una canción del karaoke.  5. Buscar la canción más fácil.  6. Buscar la canción más difícil.  7. Buscar la canción más larga. |  1. **[0.1]** Estudia el código de la clase karaoke y responde las siguientes preguntas  |  |  | | --- | --- | | Pregunta | Respuesta | | ¿A qué categoría pertenece CATEGORIAS[2] ? | CATEGORIAS[2] pertenece a la categoría de “Reggae”. | | ¿En el método darArtistasCategoria qué ocurre si se borra la instrucción auxiliar = auxiliar.darSiguiente(); ? | Si se elimina la instrucción auxiliar = auxiliar.darSiguiente();, el bucle while nunca terminará y la función se quedará en un bucle infinito. Esto se debe a que auxiliar siempre apuntará al mismo objeto, el primer artista en la lista, y nunca avanzará a través de la lista para comprobar los artistas restantes. En consecuencia, la función nunca llegará al final de la lista y nunca retornará la lista de artistas de la categoría especificada. Es importante que el puntero auxiliar se mueva a través de la lista para que la función pueda procesar todos los artistas en la lista y así retornar correctamente la lista de artistas de la categoría especificada. | | ¿En el método darCancionMasDificil qué canción se devolvería si en el if se cambia el signo a menor (<) y la variable dificultad toma el valor de 11? | Si se cambia el signo < en el if a > y dificultad toma el valor de 11, entonces la primera canción que tenga una dificultad mayor que 11 se almacenará en osCancion, en lugar de la canción con la mayor dificultad. Si hay varias canciones con una dificultad mayor que 11, se almacenará la primera canción encontrada. |  1. **[0.2]** Se requiere construir un informe con los artistas de una categoría dada por el usuario con sus canciones. El nombre reporte del reporte y su ubicación debe escogerlas usted, al igual que los datos a incluir en el informe. La situación anormal se presenta cuando no existen artistas. Complete el requerimiento funcional.  |  |  |  |  | | --- | --- | --- | --- | | ID: | RF1 | Nombre: | Generar un informe con los artistas de una categoría dada por el usuario con sus canciones. | | Resumen: | | | Se requiere construir un informe con los artistas de una categoría dada por el usuario con sus canciones. El nombre reporte del reporte y su ubicación debe escogerlas usted, al igual que los datos a incluir en el informe. La situación anormal se presenta cuando no existen artistas. | | Entradas: | | | Categoría dada por el usuario (nCategoriaUsuario) | | Resultado: | | | Un informe con los artistas de una categoría dada por el usuario con sus canciones. | | Situación anormal: | | | Si no se ingresa un nombre de categoría valido debe informarse del error.  Si no existe la categoría dada por el usuario debe informarse del error.  Si no existen artistas en la categoría dada por el usuario debe informarse del error.  Si no existen canciones en los artistas en la categoría dada por el usuario debe informarse del error. |  1. **[1.0]** Programe el método que resuelve el requerimiento funcional en la clase correspondiente. 2. **[0.2]** Asocie la funcionalidad del anterior método al metodo1 para que informe al usuario que el reporte de artistas de categorías rock o Pop con sus canciones se generó correctamente. 3. **[0.2]** Organice el código de los métodos de los numerales 5 y 6 de este laboratorio 4. **[0.3]** Documente de manera completa el método programado en el numeral 5 5. **[0.2]** Se requiere saber si existen o no canciones cortas y difíciles que pertenezcan a artistas de 2 categorías dadas por el usuario. Una canción es corta si dura menos de 3 minutos y es difícil si tiene una dificultad que sobrepasa el valor de 10. La situación anormal se presenta cuando no existen artistas. Complete el requerimiento funcional.  |  |  |  |  | | --- | --- | --- | --- | | ID: | RF2 | Nombre: | existen o no canciones cortas y difíciles que pertenezcan a artistas de 2 categorías dadas por el usuario. | | Resumen: | | | Se requiere saber si existen o no canciones cortas y difíciles que pertenezcan a artistas de 2 categorías dadas por el usuario. Una canción es corta si dura menos de 3 minutos y es difícil si tiene una dificultad que sobrepasa el valor de 10. La situación anormal se presenta cuando no existen artistas. | | Entradas: | | | Categorías (nCategoria1, nCategoria2) | | Resultado: | | | Si o no, dependiendo de si existen o no canciones cortas y difíciles que pertenezcan a artistas de 2 categorías dadas por el usuario. | | Situación anormal: | | | Si no existe la categoría ingresada por el usuario debe informase del error.  Si no existen artistas dentro de la categoría debe informarse del error.  Si no existen canciones en el artista debe informase del error. |  1. **[1.0]** Programe el método que resuelve el requerimiento funcional en la clase correspondiente 2. **[0.2]** Asocie el método anterior con el metodo2 para que muestre al usuario si existen o no canciones con las características dadas. 3. **[0.2]** Organice el código de los métodos de los numerales 10 y 11 de este laboratorio 4. **[0.3]** Documente de manera completa el método programado en el numeral 10 5. **[1.0] Seleccione uno de los problemas resuelto en este laboratorio y explíquelo mediante un video de no más de 10 minutos. Debe describir la manera en que programó tanto el método construido como su llamado en el metodo1 o metodo2. Comparte tu video con la dirección amartinez@umariana.edu.co y escriba en el siguiente espacio el link de su video**  |  | | --- | | https://drive.google.com/file/d/13NVjWKiJ67A302LyCngZCy-w1pKCvvS1/view?usp=sharing |   10. Incluye este documento debidamente diligenciado en la carpeta docs/specs de la aplicación, comprime tu ejercicio y envíalo por la plataforma virtual. | | |