

**Universidad San Carlos de Guatemala**  
**Facultad de Ingeniería**  
**Escuela de Ciencias y Sistemas**  
**Introducción a la Programación y Computación 2**

**Ing. Jaime Francisco Yuman Ramirez**  
**Tutor de curso: Monica Raquel Calderon Muñoz**



## **PROYECTO 1**

### **OBJETIVO GENERAL**

Se busca que el estudiante sea capaz de desarrollar una solución integral que implemente tipos de datos abstractos (TDA) bajo el concepto de programación orientada a objetos.

### **OBJETIVOS ESPECÍFICOS**

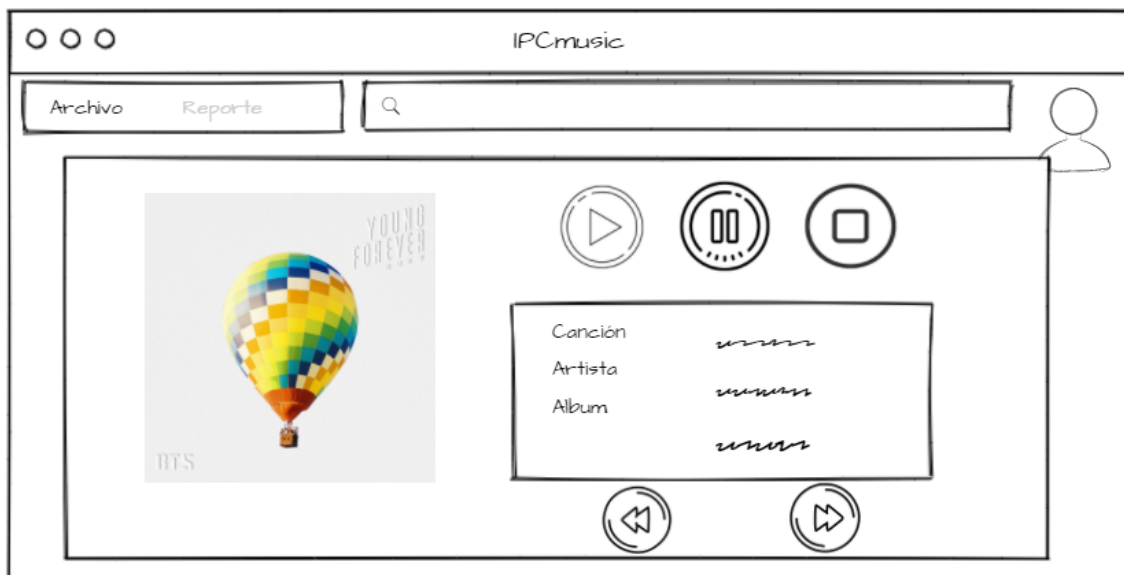
- Implementar POO para el desarrollo de la solución a través de lenguaje Python
- Utilizar estructuras de programación secuenciales, cíclicas y condicionales.
- Utilizar listas ordenadas para implementar una matriz dispersa utilizando memoria dinámica.
- Visualizar TDA'S por medio de la herramienta Graphviz.
- Utilizar archivos XML como insumos para la lógica y comportamiento de la solución.
- Utilizar código HTML para mostrar los resultados de la información obtenida y trabajada.

## DESCRIPCIÓN GENERAL

La aplicación de escritorio IPCmusic consiste en un reproductor de música el cual cuenta con una interfaz de usuario amigable e intuitiva que permitirá al usuario ordenar la música que encuentra en su ordenador, así como mostrar estadísticas con relación a sus canciones.

Dado que es un reproductor de música, se debe de poder visualizar la canción que se está reproduciendo con su respectivo artista y al álbum que pertenece en conjunto con su imagen. Así también, el reproductor debe de ser capaz de pausar una canción o detenerla por completo o bien adelantar o regresar.

***Nota:*** Dado sea el caso que una canción sea un single y no posea álbum, la información del álbum se deja vacío y de imagen se puede colocar una del artista o la propia del single



Fuente: Elaboración propia

Entre otras de las opciones que permite la aplicación realizar al usuario se encuentran:

- Cargar biblioteca de canciones a través de un archivo XML
- Crear listas de reproducción seleccionando los datos de la biblioteca (ya sea por canciones o por artistas) y reproducirlas utilizando estructuras de datos lineales.
- Mostrar un reporte de las canciones y artistas más reproducidos en HTML
- Mostrar un reporte de la biblioteca a través de la herramienta Graphviz

## IMPLEMENTACIÓN

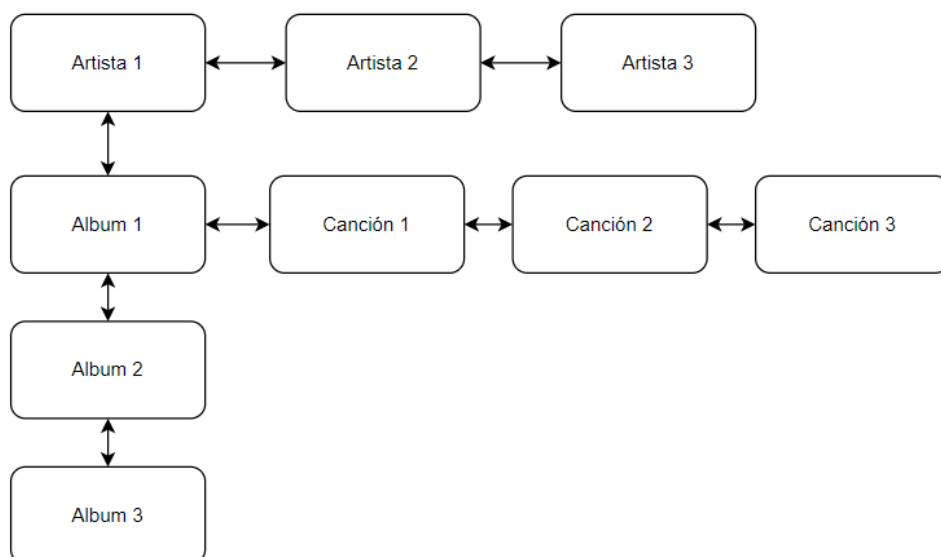
### CARGA DE BIBLIOTECA

La aplicación contará con una opción que permitirá que la biblioteca sea cargada a través de un archivo XML el cual estará formado por la siguiente estructura

```
<?xml version="1.0" encoding="UTF-8"?>
<biblioteca>
  <cancion nombre="Attack on bangtan">
    <artista>"BTS"</artista>
    <album>"O!RUL8,2?"</album>
    <imagen>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\AOB_BTS.jpg"</imagen>
    <ruta>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\AOB_BTS.mp3"</ruta>
  </cancion>
  <cancion nombre="the middle">
    <artista>"Jimmy Eat World"</artista>
    <album>"Bleed American"</album>
    <imagen>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\theMiddleJEW.jpg"</imagen>
    <ruta>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\theMiddleJEW.mp3"</ruta>
  </cancion>
  <cancion nombre="Perfectly Perfect">
    <artista>"Simple plan"</artista>
    <album>"Taking One for the Team"</album>
    <imagen>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\SimplePlan_PP.jpg"</imagen>
    <ruta>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\SimplePlan_PP.mp3"</ruta>
  </cancion>
</biblioteca>
```

Fuente: Elaboración propia

En donde los artistas, los álbumes y las canciones serán almacenados en listas doblemente enlazadas; Tanto la ruta de la canción como la imagen del álbum serán atributos de las listas de las canciones y de los álbumes respectivamente. Estas listas deben de ir conectadas de manera que, cada lista de artista contenga la lista de álbumes y cada álbum posea su lista de canciones.



Fuente: Elaboración propia

## LISTAS DE REPRODUCCIÓN

El usuario será capaz crear las listas de reproducción que desee, para ello podrá seleccionar las canciones por agregar para luego darle un nombre a la lista. Esta lista tendrá un nombre único y no podrá repetirse, si el usuario llega a ingresar el nombre de una lista ya existente debe de mostrar un mensaje de alerta y permitir que ingrese un nuevo nombre.

Las listas de reproducción serán representadas por listas doblemente enlazadas circulares, a considerar que entre los atributos que debe de poseer cada nodo de la lista debe encontrarse la cantidad de veces que una canción ha sido reproducida. Cada lista de reproducción va a poder ser reproducida por dos modos: *Normal* y *aleatorio*

- Normal:

Las funciones de una lista de reproducción en modo normal son:

1. La lista empieza a ser reproducida desde la primera canción almacenada en la lista doblemente enlazada circular.
2. Al darle al botón *atrás* y está sonando la primera canción de la lista ***la canción que debe de sonar es la última canción de la lista.***
3. Al darle al botón *siguiente* y está sonando la última canción de la lista ***la canción que debe de sonar es la primera canción de la lista.***

- Aleatorio

Las funciones de una lista de reproducción en modo aleatorio son:

1. La lista empieza a ser reproducida desde cualquier canción almacenada en la lista doblemente enlazada circular.
2. Al darle al botón *atrás* se debe de reproducir la última canción escuchada, para almacenar las canciones que se han ido escuchando se recomienda el uso de una pila ***y será la única estructura de datos que puede utilizar a partir de las librerías propias de python.***
3. Al darle al botón *siguiente* se continúa con el mismo principio de aleatoriedad, por lo que debe de reproducir cualquier canción de la lista doblemente enlazada circular ***a excepción de la canción que se está escuchando actualmente.***

## GUARDANDO LAS LISTAS DE REPRODUCCIÓN

Las listas de reproducción deben de poder ser almacenadas en un archivo XML para que, al momento en que en reproductor sea nuevamente abierto estás puedan ser cargadas desde cero.

**Nota:** Únicamente es necesario almacenar la información de las listas, no se debe de almacenar información al respecto de la canción que se estaba reproduciendo al momento de guardar la lista.

El archivo XML para almacenar la información de las listas de reproducción está formado por la siguiente estructura

```
<?xml version="1.0" encoding="UTF-8"?>
<ListasReproduccion>
  <Lista nombre="MiLista">
    <cancion nombre="Attack on bangtan">
      <artista>"BTS"</artista>
      <album>"O!RUL8,2?"</album>
      <vecesReproducida>2</vecesReproducida>
      <imagen>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\AOB_BTS.jpg"</imagen>
      <ruta>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\AOB_BTS.mp3"</ruta>
    </cancion>
    <cancion nombre="the middle">
      <artista>"Jimmy Eat World"</artista>
      <album>"Bleed American"</album>
      <vecesReproducida>3</vecesReproducida>
      <imagen>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\theMiddleJEW.jpg"</imagen>
      <ruta>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\theMiddleJEW.mp3"</ruta>
    </cancion>
    <cancion nombre="Perfectly Perfect">
      <artista>"Simple plan"</artista>
      <album>"Taking One for the Team"</album>
      <vecesReproducida>7</vecesReproducida>
      <imagen>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\SimplePlan_PP.jpg"</imagen>
      <ruta>"C:\Users\Monica Calderon\Desktop\Diciembre 2021\IPC2\SimplePlan_PP.mp3"</ruta>
    </cancion>
  </Lista>
</ListasReproduccion>
```

Fuente: Elaboración propia

## REPORTES

El reproductor de música debe de ser capaz de mostrar dos diferentes tipos de reporte:

### 1. Reporte HTML

Presentará de manera gráfica las canciones que más han sido reproducidas así como a la lista de reproducción a la que pertenecen.

### 2. Reporte Graphviz

Se deberá poder visualizar en cualquier momento de la ejecución de la aplicación el estado en el que se encuentran las estructuras de datos.

## INTERFAZ GRÁFICA

El diseño de la interfaz gráfica queda a discreción del estudiante. Se tomará en cuenta lo intuitivo así como la presentación que tenga.

## CONSIDERACIONES

Debe utilizarse versionamiento para el desarrollo del proyecto. Se utilizará la plataforma Github en la cual se debe crear un repositorio en el que se gestionará el proyecto. Se deben realizar mínimo 4 releases o versiones del proyecto. Se deberá agregar a su respectivo auxiliar como colaborador del repositorio. El último release será el release final y se deberá de realizar antes de entregar el proyecto en la fecha estipulada.

Para la realización de la interfaz gráfica queda a discreción del estudiante que librería utilizar.

## DOCUMENTACIÓN

Para que el proyecto sea calificado, el estudiante deberá entregar la documentación utilizando el formato de ensayo definido para el curso. En el caso del proyecto, el ensayo puede tener un mínimo de 4 y un máximo de 7 páginas de contenido, este máximo no incluye los apéndices o anexos donde se pueden mostrar modelos y diseños utilizados para construir la solución. Este informe debe expresar con claridad el diseño de objetos ideado para resolver este proyecto por lo que debe expresar el diagrama de clases y los diagramas de actividades de los algoritmos más importantes.

Debe de tomar en cuenta que es un ensayo formal, por lo que se calificará tanto la redacción, como ortografía y presentación.

## RESTRICCIONES

- Solo se permitirá la utilización de los IDEs discutidos en el laboratorio.
- Las listas deben ser creadas completamente por el estudiante mediante clases. Caso contrario, se penalizará con el 100% de la nota.
- Uso obligatorio de programación orientada a objetos (POO).
- El nombre del repositorio debe de ser IPC2\_Proyecto1Diciembre\_#Carnet.
- El estudiante debe entregar la documentación solicitada para poder optar a la calificación.
- Los archivos de entrada no podrán modificarse.
- Se calificará la versión del cuarto release o del último release realizado previo a la fecha de entrega. No se calificará dado se dé el caso que existan modificaciones de código en fechas posteriores a la entrega.
- Para dudas concernientes al proyecto se utilizarán los foros en UEDI de manera que todos los estudiantes puedan ver las preguntas y las posteriores respuestas.
- De no existir una forma gráfica de poder validar el funcionamiento de la aplicación se penalizará con el 100% de la nota.
- **COPIAS TOTALES O PARCIALES SERÁN REPORTADOS A LA ESCUELA Y OBTENDRÁN NOTA DE 0 PUNTOS.**
- **NO HABRÁ PRÓRROGA.**

## ENTREGA

- La entrega será el día viernes 17 de diciembre antes de las 23:59.
- La entrega será por medio de la UEDI.