ANALISIS

Para obtener el diseño de la aplicacion primero se redujo el enunciado en apartados (componentes) con un objetivo especifico. Luego, se describio los requisitos de cada apartado, las condiciones en las que funciona y los resultados que devuelve. Es asi como se obtuvo lo siguiente:

spotify
- en c++ para linux
- arhivo de entrada .xml
- app en terminal

opciones
- operaciones de canciones
  - insertar: ingresa canciones <al sistema>
    - solicitar nombre,path hasta que sean correctos
  - eliminar: elimina canciones <del sistema>
    - solicitar nombre (string) o indice (int)
    - confirmar eliminacion mostrando nombre,path
  - buscar cancion en la store por nombre
    - solicitar nombre
    - mostrar (si existe la cancion) nombre,path,indice
  - listar canciones <del sistema> mostrando id,nombre,path
- operaciones de playlist
  - crear playlist
    - solicitar nombre,descripcion
    - solicitar agregar canciones
  - eliminar playlist
    - mostrar playlists
    - solicitar id de playlist, luego eliminar por id
  - actualizar playslit
    - solicitar id de playlist luego solicitar datos nuevos para nombre,descripcion
    - si algun campo se ingresa vacio entonces para ese campo sus datos no se modifican
  - listar playlists:
    - muestra las playlists <del sistema> mostrando <indice>. <nombre> - <descripcion>
    - solicitar indice y mostrar las canciones de esa playlist
  - agregar canciones
    - mostrar canciones de la store
    - solicitar indice de cancion, luego se agrega a esta playlist
  - eliminar canciones
    - mostrar canciones de esta playlist
    - solicitar indice de una cancion, luego se elimina de esta playlist
- reproduccion
  - normal
    - orden: mismo orden que el de la playlist, se detiene despues de ultima cancion
    - estructura: lista doble enlazada
  - repetir
    - orden: mismo orden que el de la playlist, despues de la ultima cancion sigue la primera
    - estructura: lista doble enlazada circular
  - siguiente
  - anterior
  - ver playlist: muestra las proximas canciones a reproducir
  - agregar una cancion a una playlist prioritria
    - muestra la store y solicita el indice de una cancion
    - la cancion se agrega a la playlist prioritaria
    - playlist prioritaria
      - cada playlist debe tener una playlist prioritaria
      - la estructura de la playlist prioritaria es una pila
      - la playlist prioritaria se reproduce completamente antes de seguir con la playlist seleccionada
- carga masiva con un archivo .xml
  - elementos del xml
    - <Insertar>

```
                                    <cancion>
                                            <Nombre>
                                            <path>
                                    <cancion>
                                            <Nombre>
                                            <path>
                                            <Pos> <!-- supongo: posicion en la store -->
                                    <Lista>
                                            <Nombre>
                                            <Descripcion>
                                    <Lista>
                                            <Nombre>
                                            <Descripcion>
                                            <Canciones> <!-- ingresar lista con canciones -->
                                                    <pos> <!-- indice de cancion en la store -->
                            <Eliminar>
                                    <cancion>
                                            <id>
                                    <cancion>
                                            <Nombre>
                                    <Lista>
                                            <id>
                                    <Lista>
                                            <Nombre>
                                    <Lista>
                                            <id>
                                            <canciones>
                                                    <pos>
        salir
store
        es la playlist principal
reproduccion
        solicitar el indice de una playlist, luego reproducir

playlist
        no tienen limite de caciones
        su orden es segun se introduzcan las canciones
        el identificador/indice de una cancion es la posicion que tiene en la playlist
```

Posteriormente, a partir de lo anterior se creo el menu de opciones con una descripcion breve de lo que producen. Lo cual es:

```
ESTRUCTURAS DE DATOS
lista doble enlazada
   agregar nodo
      por su posicion
      al final de la lista
   buscar nodo
      por su posicion
      por su dato
   eliminar nodo
      por su posicion
      por su dato
PILA
   agregar nodo
   eliminar nodo
```

<<<<<<<<<<<<<< SPOTIFY >>>>>>>>>>>>>>
reproduccion
    reproducir playlist
        modo normal: lista doble enlazada
        modo repetido: lista doble enlazada circular
    pausar/continuar
    anterior
    siguiente
    proximas canciones: pila
    agregar a proximas -> mostrar store
store
    mostrar canciones de store: imprimir(posicion,nombre,path)
    buscar cancion en store: solicitar(nombre)
    ingresar canciones: solicitar(nombre,path)
    eliminar canciones: solicitar(posicion|nombre)
playlists
    mostrar playlists: imprimir(posicion,nombre,descripcion)
    abrir playlist
        mostrar canciones de una playlist: solicitar(posicion)
        actualizar playlist: solicitar(posicion,nombre?,descripcion?)
        agregar canciones a playlist: solicitar(posicion)
        eliminar canciones de playlist: solicitar(posicion)
    modificar playlists
        crear playlist: solicitar(nombre,descripcion),agregarCanciones()
        eliminar playlist: solicitar(posicion)
importar configuracion