INFORME PREELIMINAR DESAFIO 2.

Para este desafío se nos pide modelar una plataforma de streaming llamada “UdeaTunes”, utilizando el paradigma de la programación orientada a objetos, donde se plantee un funcionamiento eficiente en cuanto a procesamiento y memoria.

Ahora para contextualizar de mejor manera los requerimientos del programa los voy a mencionar a manera de listado y si es necesario más adelante lo explicare cuando muestre mi análisis y planteamiento de la solución:

* Carga / actualización de datos.
* Ingreso a la plataforma.
* Reproducción aleatoria.
* Lista de favoritos:

Editar, seguir y ejecutar.

* Medición consumo de recursos.

Ahora teniendo un mejor panorama del problema a resolver vamos a empezar con mi análisis y planteamiento de la solución. La mejor manera de generar un análisis y plantear una solución seria en primer lugar entender en su totalidad lo que se nos pide hacer, tras tener esto en cuenta podemos empezar a desarrollar un diagrama de clases de manera abstracta donde no nos centremos tanto en los atributos y métodos de las clases, sino, en sus definiciones y relaciones entre ellas. En este caso eso fue lo que yo hice y voy a explicarlo de la manera más entendible y concisa, para finalmente adjuntar una imagen donde se pueda apreciar el diagrama de clases.

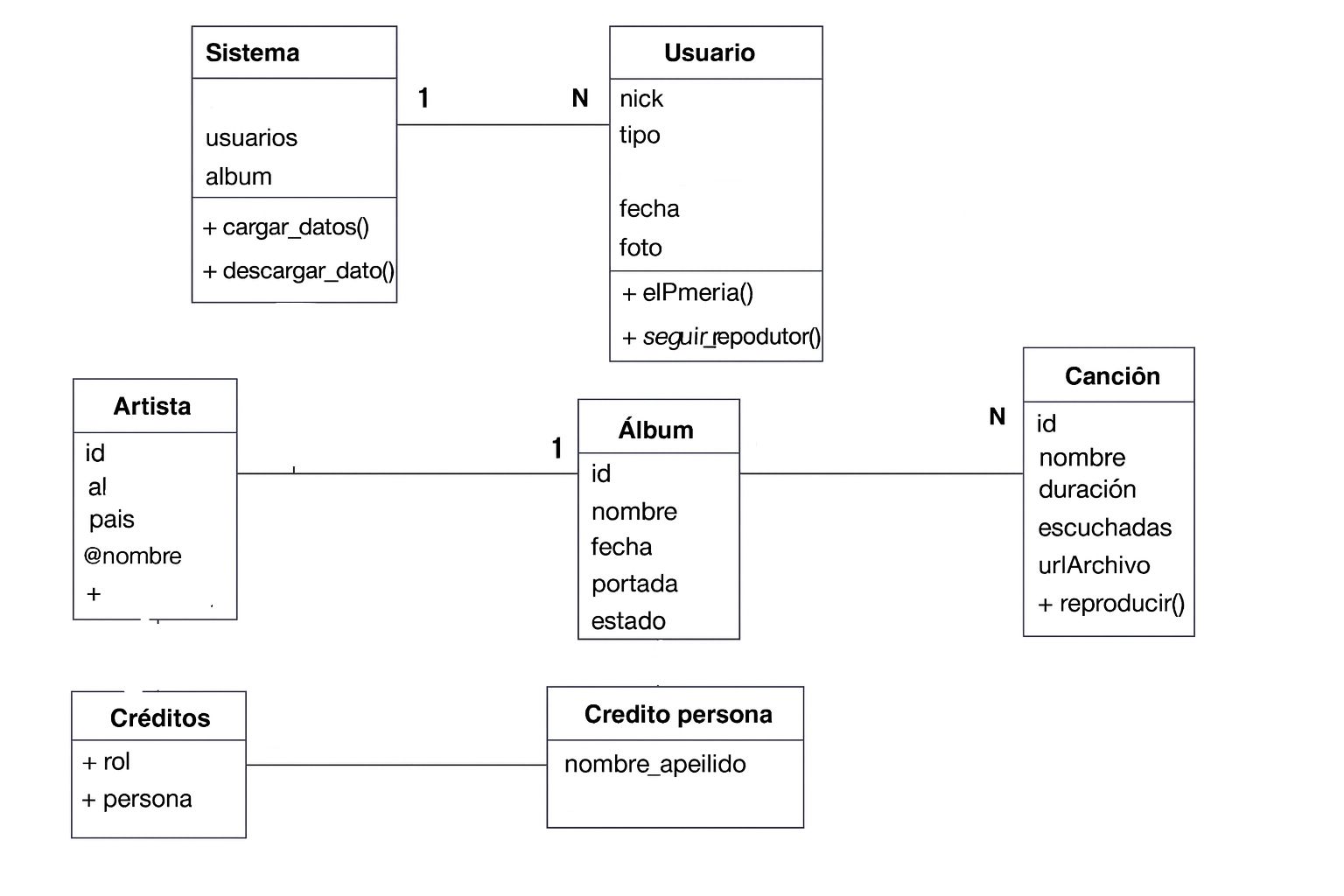
En mi planteamiento en primer lugar definí la clase “Sistema” de donde vamos a partir para el resto de clases. En esta clase proponemos la solución para la carga / actualización de datos y el login a la plataforma.

También plantee la clase “Usuario” para entre otras funcionalidades, poder diferenciar entre estándar o premium, esta clase tiene relación 1 a N con la clase Sistema.

Ahora tenemos una agrupación, donde a la izquierda del todo tenemos la clase “Artista” con sus respectivos atributos, tales como, id, edad, país, etc. Luego la clase “Álbum” con relación 1 a N con la clase artista. Finalmente, la clase “Canción” con relación 1 a N con la clase álbum.

Para el problema que nos generan los créditos plantee 2 clases “Creditos” que va a ser la agrupación de productores, músicos y compositores de una canción y “CreditosPersona” haciendo referencia a cada persona de los créditos y su código de afiliación.

Todo esto se vería algo así:



Finalmente tenemos otras clases no relacionada, takes como:

“RutasMedia” para las rutas de las canciones, porque hay 2 tipos.

“MensajePublicidad” porque cada mensaje tiene una probabilidad.

“ListaFavoritos” para poder implementar las funcionalidades relacionadas a la lista.

“Reproductor” para implementar las funciones de reproducción.