Musical Jukebox

Vague, high-level problem statement, as delivered in an interview: Design/Model a musical jukebox

What that typically means:

Design a simple jukebox that plays different types of media. Make simplifying assumptions to begin with, e.g. assume that there is only one disk-type, and that it has a few songs. Extend it to have multiple media types and even playlists.

"A true jukebox refers to a machine that plays songs derived from a collection of music held within the unit itself. This music can be stored on all types of media format including records, Cds and more recently digital. In both a home and coin operated scenario the jukeboxes play songs selected by pressing a combination of buttons pinpointing the location and track number in relation to Records and Cds or the album name and tack number when dealing with digital music."

If you have not seen a Jukebox, this is what it looks like: https://www.youtube.com/watch?v=f2GBNafHceY

Deliverables:

1. A class diagram, showing relationships with each other where appropriate. Classes should show state and methods. Use any convenient notation. UML is more widely known.

2. Main() method, showing how you'll initialize your system and start using it.

3. Identify the design pattern.

API:

API here is quite simplistic. It's mostly READ and variants of READ.

A more interesting extension to design, is an online music service
e.g. Spotify or SoundCloud.