Brandon Baither
CIDM 6330

Spotify Playlist Generator

Entities and attributes:

- **Playlist** — Spotify ID, PlaylistID, Name
- **Contains** — Position
- **Song** — SongID, Artist, Title, Album, Spotify URI
- **User** — Spotify User ID
- **Tagged** — Value
- **Rule Set** — Name, RuleSetID, Song Count
- **Rule** — Requirement, Threshold, Count, Percent, Sort
- **Tag** — Name, TagID, isNumeric

Relationships: Builds, Listens To, Generates, Defines, Labels

Cardinalities:
- Playlist N — Contains — N Song
- Playlist N — Builds
- Song N — Listens To
- User 1 — Builds, User 1 — Listens To
- User 1 — Defines — N Rule Set
- User 1 — Labels — N Tag
- Playlist N — Generates
- Rule Set 1 — Generates
- Rule Set N — Rule — N Tag
- Song N — Tagged — N Tag

Uncertain at this point if it will be necessary/valuable to keep a separate userID or just have the user login with their Spotify ID. Strongest reason I've come up with so far is to use this to move songs/playlists between multiple Spotify accounts, but not sure if that's a complexity I want to support.

The Rules/Rulesets got a bit complicated. Essentially two modes here, with probably a checkbox/drop down to switch between. In one you set the number of songs to go in the playlist ("Song Count") then define percentages in the Rules. In the other you set counts in the rules and the total is just the sum of those. Requirement is how you relate the inclusion/exclusion to the Tag - ie contains/does not contain for string tags, or greater than/equal to/etc for numeric tags - and the threshold is to define that value for numerics.

Color Key
Required Attribute
Optional Attribute