Project 2 - Workplace Music Playlist Generator

Team Name: PlaylistNRG

Team Members: Christian Randle, Matthias Quintero, Nicholas Desmornes, Jake Willis

Team Lead: Matthias Quintero

Gitflow Manager: Christian Randle

Topic Proposal:

Our team will design an app which uses information gathered from the spotify api to generate work safe playlists. Employees will be able to log into our app using their Spotify credentials, where a manager of a given shift can then aggregate a playlist based off of their favorite tracks/artists. The manager will also be able to view the statistics used from their shift group to generate the playlist and be able to accept or deny song requests from employees. The playlist can be further customized by the manager by adding filters for explicit songs or undesired categories (artists/genres).

User Stories:

1. Employees are able to log into the app using their Spotify accounts.
2. Employees are able to logout of the app.
3. Employees are able to submit song requests.
4. Employees are able to see individual statistics and team statistics.
5. Managers are able to log into the app using their Spotify accounts.
6. Managers are able to logout the app.
7. Managers are able to see their individual and shift groups aggregate statistics.
8. Managers are able to generate a playlist
9. Managers are able to approve/deny Employee song requests
10. Managers are able to blacklist songs

Used APIs

-https://developer.spotify.com/documentation/web-api/

SCHEMA

Artist

- Artist name
- Genre

Artist_Track

- Artist ID
- TrackID

Track:

- ID: Spotify ID (base 62 string identifier)
- Explicit: boolean

Employee

- Employeeid: serial value
- ManagerID: foreign key referencing managerid in Manager table
- Name
- Email
- Username
- Password

Group

- Group ID
- ManagerID

Group_Employee

- Group ID
- Employee ID

Manager<?>

- ManagerID: primary key
- Name:
- Email:
- Username:
- Password:

Notes:

- Genres are stored by artist
- A track can potentially have multiple artists and genres (track links to an array of artist objects, each of which can have an array of strings for genre's)

Assumptions

    - Will need to parse a track object containing at least (track, artist, genre) for our database
    - In the event of no overlapping artists among employees select randomly with equal distribution.

Links and Resources

- Spotify Object Models
- Getting a User's Saved tracks
- Get User's Followed Artists