Project Title; SpotifyStats

Goal Statement; Provide a aesthetic experience for viewing Spotify listening history and analytics. And also unique perspective on discovering and recommending new music to users.

Blue parallolgram; Spotify API permissions / Authentication

Purple Rectangle; Backend of our app
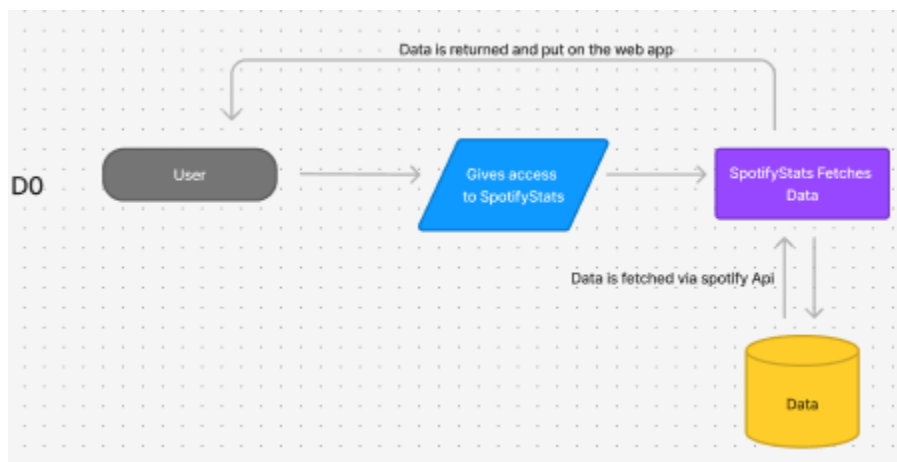
Yellow Cylinder; Database

Grey Rectangle; User

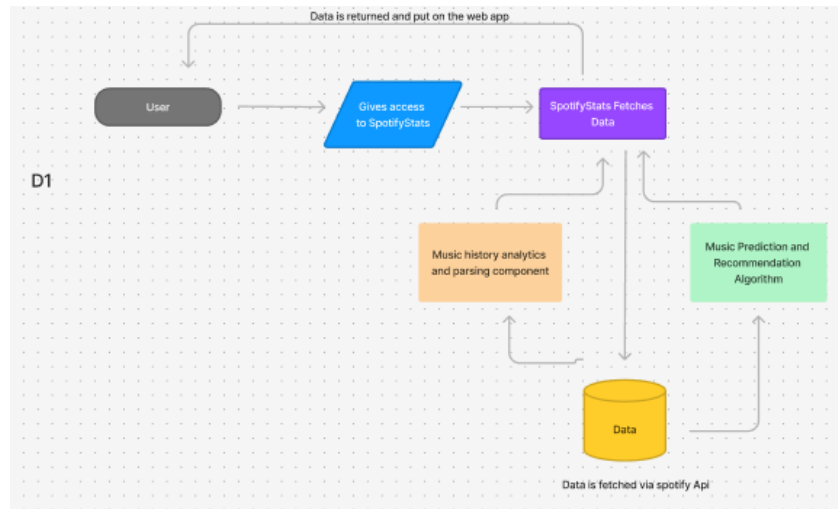Orange Box; User Analytics component

Green Box; Music Prediction Component

Green box 2; UI

D0; User connects their spotify account to our webapp and we redeliver their data to them in a clean way.

D1; A User connects their spotify account to our webapp and we use the spotify api to access their listening history. This data is then sent to 2 different components. One of which breaks down their current analytics and the other uses the data to provide song recommendation. All returned to the user via a webapp.

Diagram D1: flowchart showing Data is returned and put on the web app, User → Gives access to SpotifyStats → SpotifyStats Fetches Data, with Music history analytics and parsing component, Music Prediction and Recommendation Algorithm, Data store, and Data is fetched via spotify Api.

D2; A user connects to our webapp using the spotify api and letting us access their listening history. We are also collecting other data on general users and music to interpolate for our prediction and reccomendation algorithms. The individual users listening history is also passed to our analytics tool for breakdown and it bundled up in the backend and returned to the user on a UI.

Diagram D2: flowchart showing Data is returned and put on the web app, UI/UX, User → Gives access to SpotifyStats → SpotifyStats Fetches Data, with Music history analytics and parsing component, Music Prediction and Recommendation Algorithm, User data store, General Spotify user Data (for comparison purposes) store, and Data is fetched via spotify Api.