Spotifoo Web App

Table of Contents

1. About

Spotifoo is a web-based graphical user interface that allows you to stream music instead of downloading it from an unknown source. The user can listen to the songs according to his choice and he shall be able to choose the song based on his interest. This application is providing features to users like display all the songs, by category (album, artist and genre) and searching the songs by name. The user shall search for a particular song using search option.

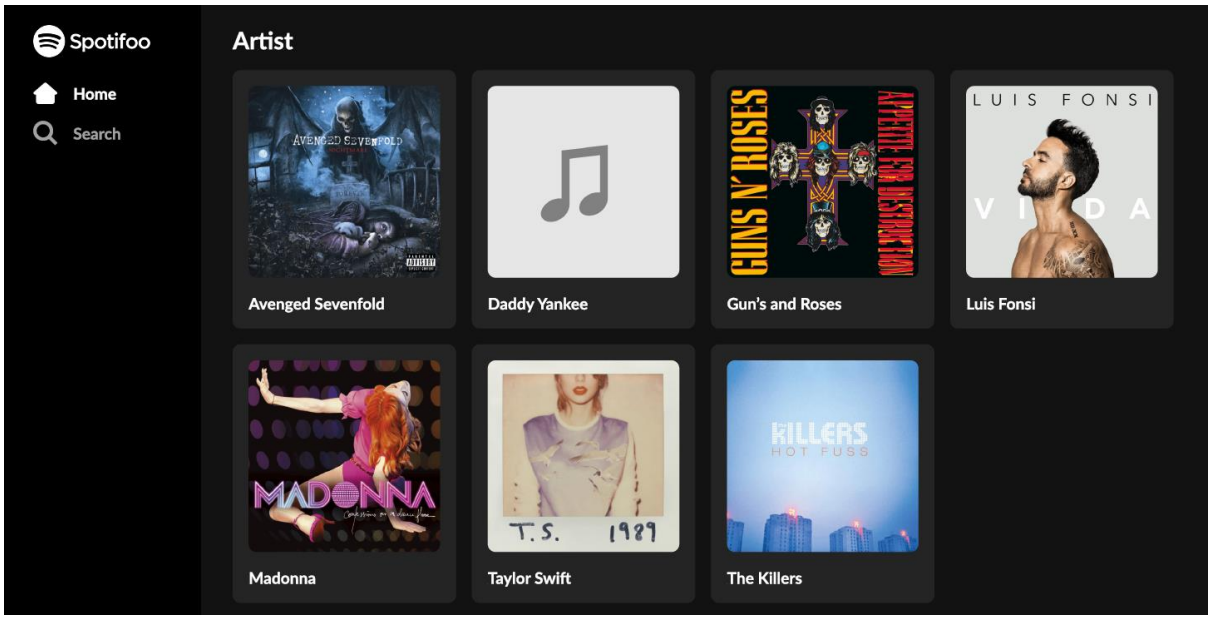
2. Project Overview

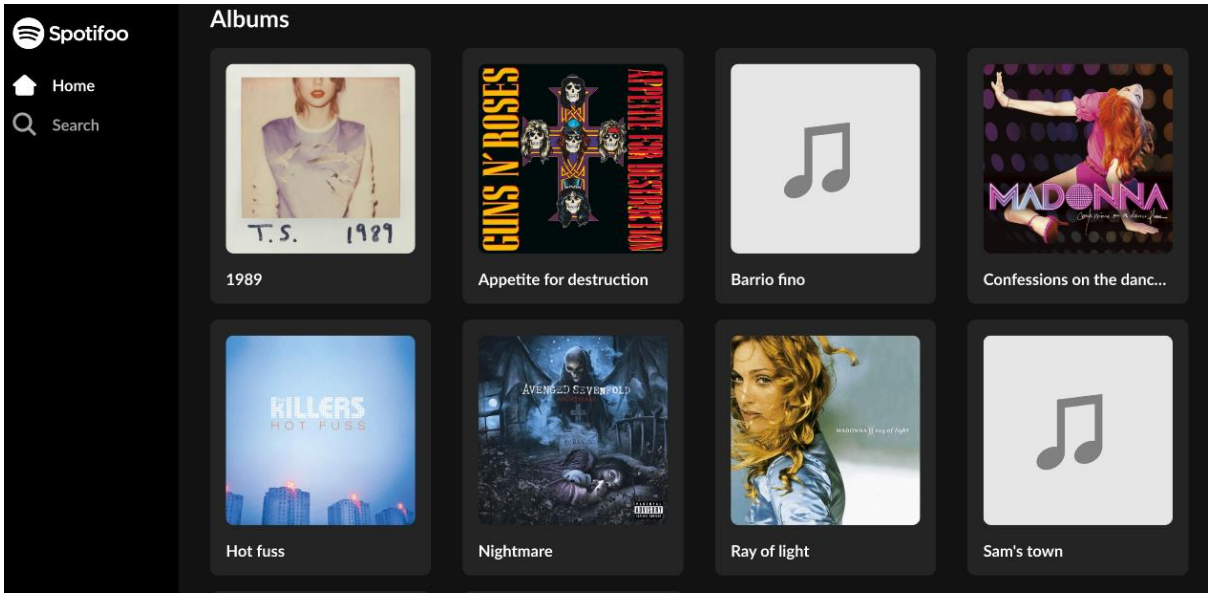
The application is having a graphical user interface. The user shall navigate through two pages .They are:

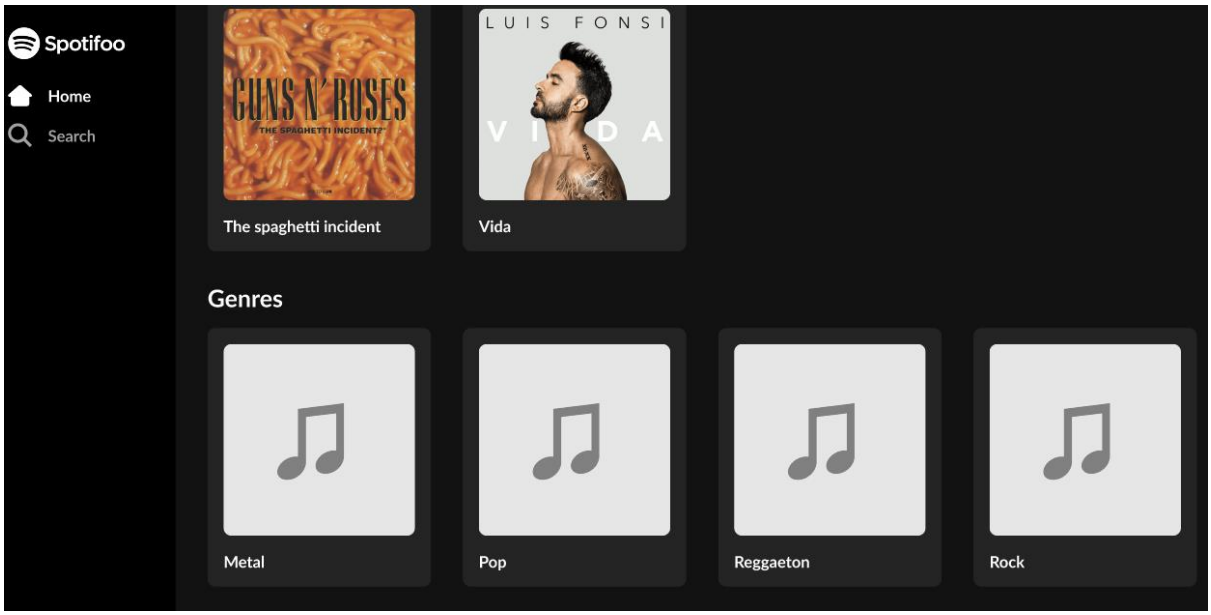
- Home
- Search

2.1 Home

When user clicks on the Home Icon, he shall see all the list of songs in the landing page. All the songs shall be divided into 3 categories. They are Artist, Album and Genre. All the songs are displayed in the landing page. The screenshot of the home page is shown below:

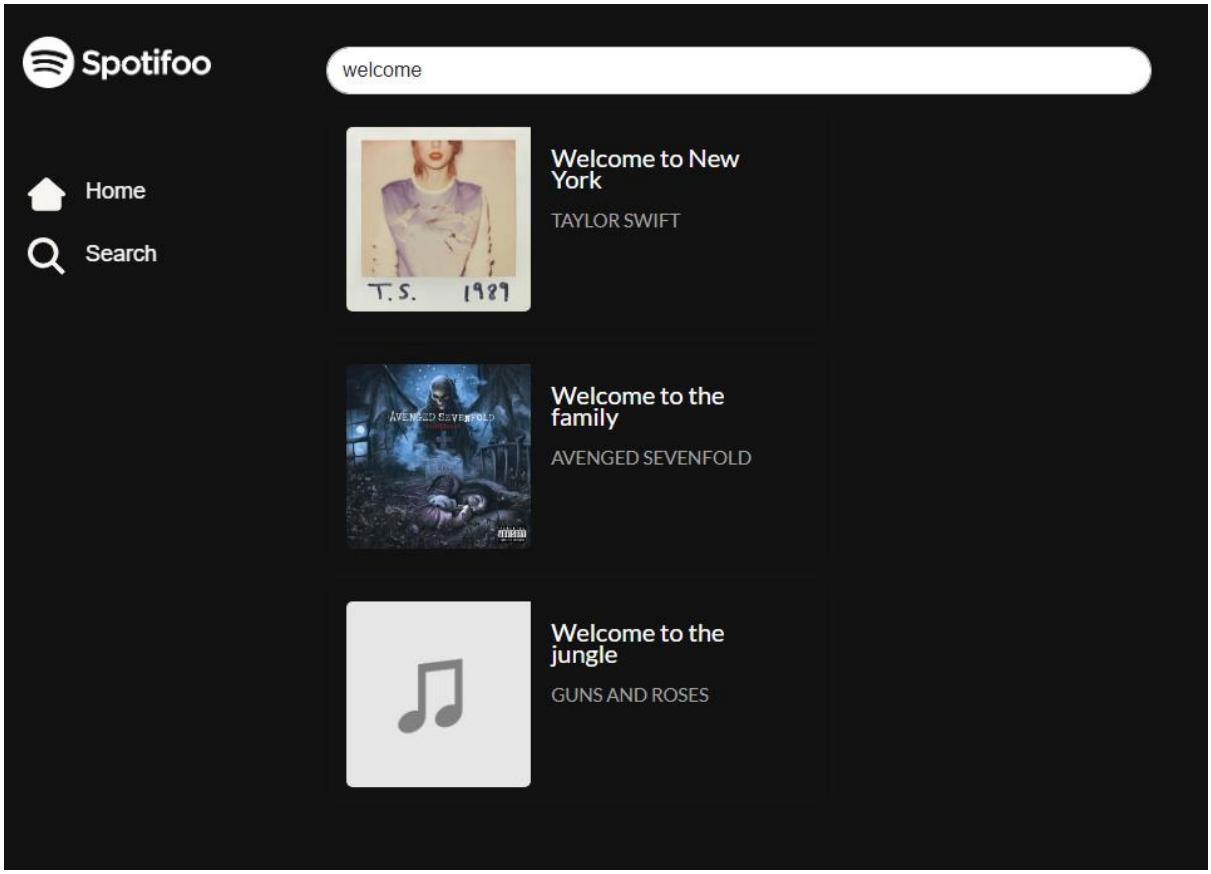






2.2 Search

When the user wants to search for a particular song in the Album, Artist or Genre, he shall click on the search icon and search for a particular song. When the user enters the title of the particular song, the song details are displayed in the page. The user shall search by the Artist ,Album and Genre. The screenshot of the search is shown below:



3. Implementation and Design

3.1 Technologies

The following technologies are used in designing and developing the Spotifoo Web App application.

1. TypeScript – Coding language for implementing the Spotifoo Web App application
2. React -Framework for building the Graphical User Interface
3. Visual Studio - IDE Purpose
4. Microsoft word – Document preparation

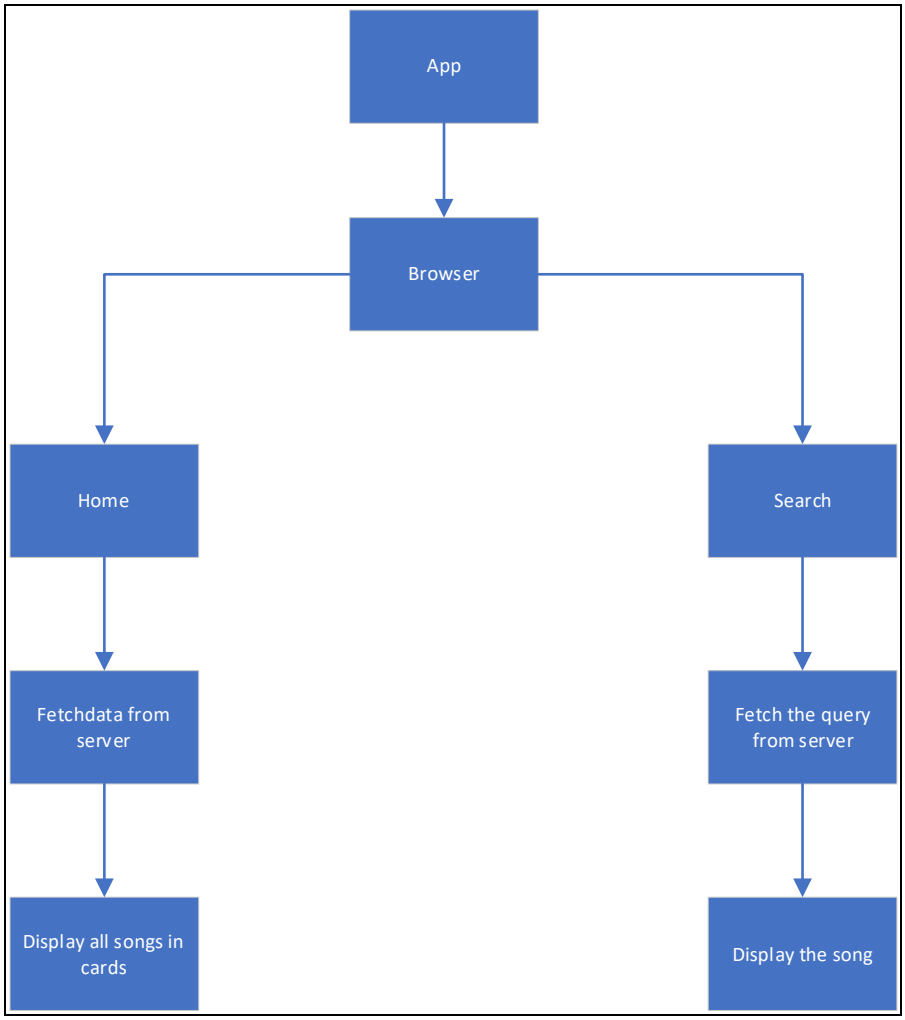
3.2 Dependencies

The dependencies that are installed in this project shall be shown in package.json file. The screenshot is shown below:

```
{
    "name": "spotifoo-web-client",
    "version": "0.1.0",
    "private": true,
    "proxy": "http://localhost:8080",
    "dependencies": {
        "@fortawesome/fontawesome-svg-core": "^6.2.0
        "@fortawesome/free-solid-svg-icons": "^6.2.0
        "@fortawesome/react-fontawesome": "^0.2.0",
        "@testing-library/jest-dom": "^5.16.5",
        "@testing-library/react": "^13.4.0",
        "@testing-library/user-event": "^13.5.0",
        "@types/jest": "^27.5.2",
        "@types/node": "^16.11.61",
        "@types/react": "^18.0.21",
        "@types/react-bootstrap": "^0.32.30",
        "@types/react-dom": "^18.0.6",
        "@types/react-icons": "^3.0.0",
        "@types/react-router-dom": "^5.3.3",
        "@types/styled-components": "^5.1.26",
        "bootstrap": "^5.2.2",
        "bootstrap-3-card": "^0.2.0",
        "react": "^18.2.0",
        "react-bootstrap": "^2.5.0",
        "react-dom": "^18.2.0",
        "react-hook-form": "^7.36.1",
        "react-icons": "^4.4.0",
        "react-router-dom": "^6.4.1",
        "react-scripts": "4.0.3",
        "reactstrap": "^9.1.4",
        "styled-components": "^5.3.6",
        "typescript": "^4.8.3",
        "web-vitals": "^2.1.4"
    },
```

3.3 Component Diagram

Component diagram is used to visualize the organization and relationships among components in the system. This diagram is used in representing the implementation of a system. The component diagram of this is shown below:


App
Browser
Home
Search
Fetchdata from server
Fetch the query from server
Display all songs in cards
Display the song