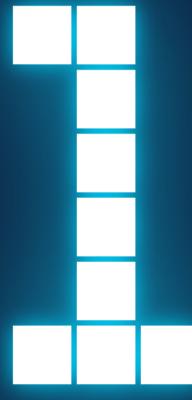




Music Place

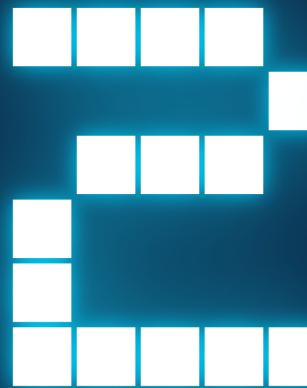
Orizio Leonardo, Diego Grazioli



Cos'è il Music Place?



Music Place è una piattaforma web che offre un vasto catalogo di oltre 10.000 brani musicali, spaziando dal 1958 fino al 2023. Ogni canzone è accompagnata da dettagliati metadati, inclusi titolo, artista, popolarità e altre informazioni rilevanti. Inoltre, Music Place offre un'esperienza interattiva unica, consentendo agli utenti di mettere alla prova le proprie conoscenze musicali attraverso giochi di memoria basati su brevi anteprime delle canzoni.

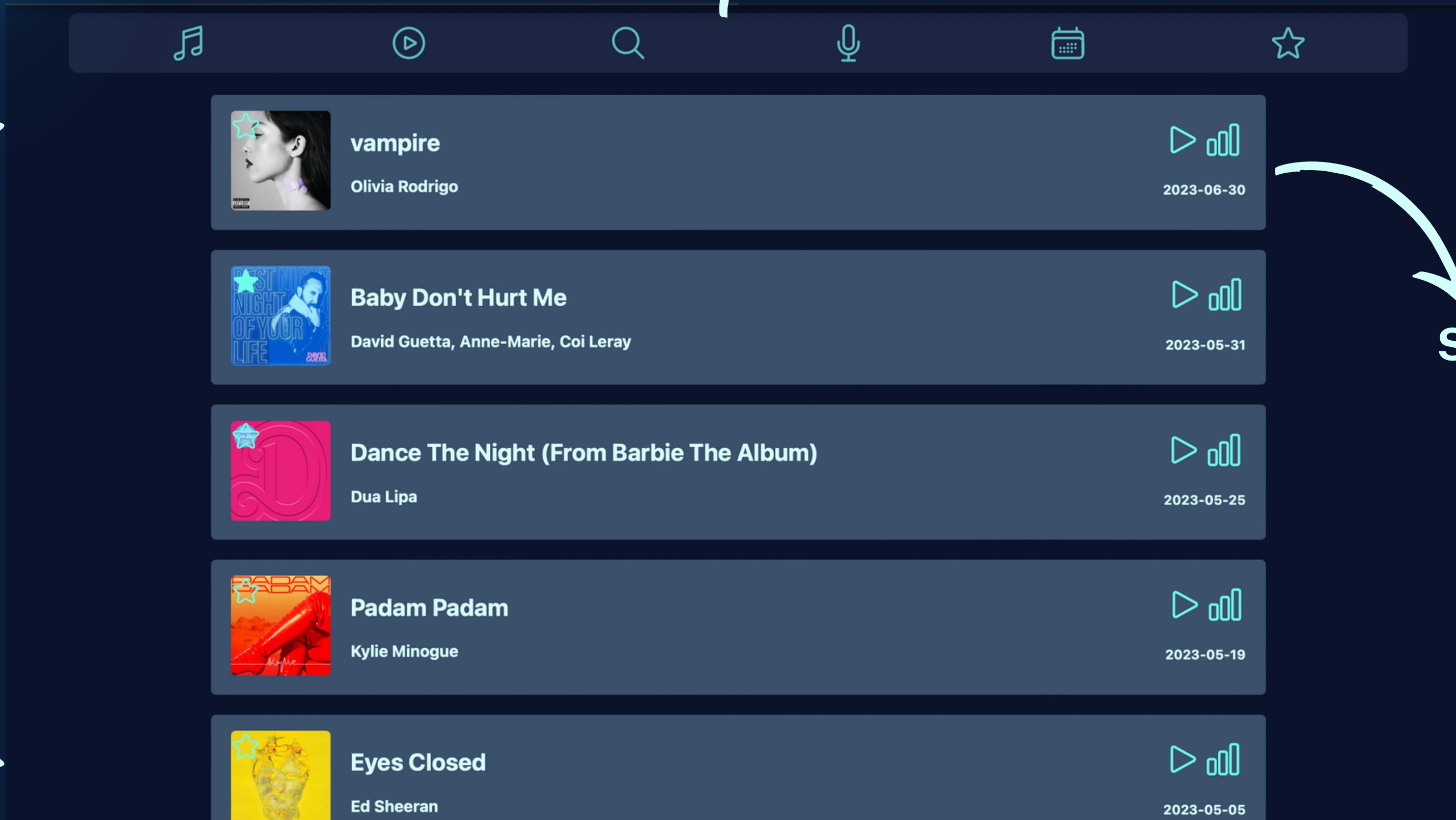


Layout



NavBar

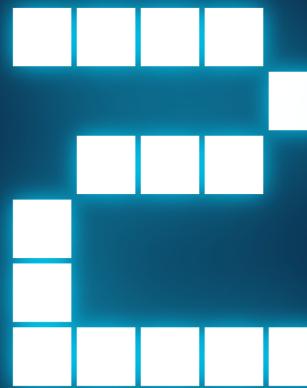
Home



Song	Artist	Date
vampire	Olivia Rodrigo	2023-06-30
Baby Don't Hurt Me	David Guetta, Anne-Marie, Coi Leray	2023-05-31
Dance The Night (From Barbie The Album)	Dua Lipa	2023-05-25
Padam Padam	Kylie Minogue	2023-05-19
Eyes Closed	Ed Sheeran	2023-05-05



SongCard



Layout

NavBar



Guess the song

Which song is this?

▶ 0:00 | 0:29

Crying for No Reason - Katy B

The Door - Silverchair

Freak Me - Silk

Ashes - Stellar

Play

GuessTheSong



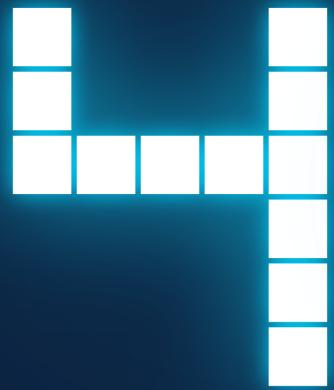
Props



```
{#each canzoni as track}
  <SongCard
    id={track['Track URI']}
    titolo={track['Track Name']}
    artisti={track['Artist Name(s)']}
    image={track['Album Image URL']}
    preview={track['Track Preview URL']}
    release_date={track['Album Release Date']}>
  </SongCard>
{/each}
```

SongCard

```
export let titolo
export let artisti
export let image
export let preview
export let release_date
export let id
```



Event dispatcher

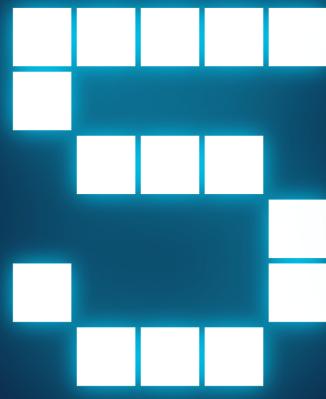
GuessTheSong

```
import { onMount, afterUpdate, createEventDispatcher } from 'svelte';
const disp = createEventDispatcher();

{#each alternatives as track}
  <button on:click={() => {
    if (track['Track URI'] === canzone_da_inovinare['Track URI']) {
      alert('Correct!');
      disp('win', "win");
    } else {
      alert('Wrong!');
      disp('lose', "lose");
    }
    selectSongAndAlternatives();
    let audio = document.getElementById('audioPlayer');
    if (audio) {
      audio.src = canzone_da_inovinare['Track Preview URL'];
      audio.load();
    }
  }}>
    {track['Track Name']} - {track['Artist Name(s)']}
  </button>
{/each}
```

Play

```
<GuessTheSong
  year={year}
  on:win={updateWinCount}
  on:lose={resetWinStreak}
  onlyFavourite = {onlyFavourite}>
</GuessTheSong>
```



Svelte store

Home

```
<script>
  import { tracks } from './store.js';
  import SongCard from './SongCard.svelte';
  let canzoni = $tracks;
</script>
```

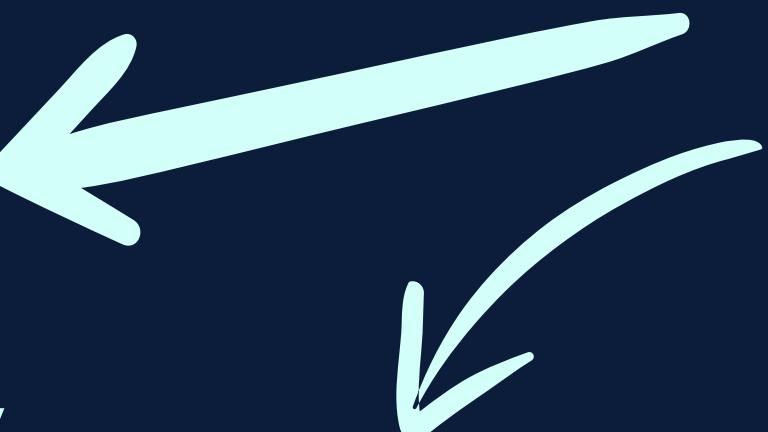
```
<main>
  <div class="tracks_container">
    {#each canzoni as track}
      <SongCard
        id={track['Track URI']}
        titolo={track['Track Name']}
        artisti={track['Artist Name(s)']}
        image={track['Album Image URL']}
        preview={track['Track Preview URL']}
        release_date={track['Album Release Date']}></SongCard>
    {/each}
  </div>
</main>
```

Play

```
import { winCount, winStreak} from './store.js';
let wins = $winCount;
let streak = $winStreak;

function updateWinCount(event) {
  streak += 1;
  if (streak > wins) {
    wins = streak;
    winCount.update(n => wins);
  }
  winStreak.update(n => streak);
}

function resetWinStreak(event) {
  if (streak > wins) {
    wins = streak;
  }
  streak = 0;
  winStreak.update(n => streak);
  winCount.update(n => wins);
}
```



store.js

```
import { writable } from 'svelte/store';
import json_data from './data/top_10000_songs_spotify.json';

function confronto(a, b) {
  if (a['Album Release Date'] > b['Album Release Date']) {
    return -1;
  }
  if (a['Album Release Date'] < b['Album Release Date']) {
    return 1;
  }
  return 0;
}

json_data.sort(confronto);

const storedWinCount = JSON.parse(localStorage.getItem('winCount'));
const storedTracks = JSON.parse(localStorage.getItem('myTracks'));
const storedWinStreak = JSON.parse(localStorage.getItem('winStreak'));

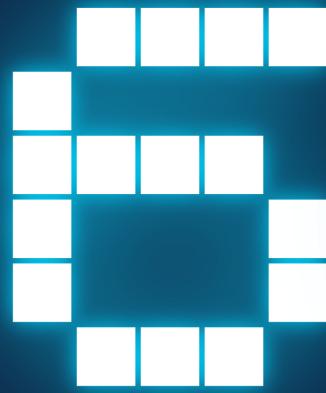
export let tracks = writable(json_data);
export let myTracks = writable(storedTracks || []);

export let winCount = writable(storedWinCount || 0);
export let winStreak = writable(storedWinStreak || 0);

winCount.subscribe(value => {
  localStorage.setItem('winCount', JSON.stringify(value));
});

winStreak.subscribe(value => {
  localStorage.setItem('winStreak', JSON.stringify(value));
});

myTracks.subscribe(value => {
  localStorage.setItem('myTracks', JSON.stringify(value));
});
```



afterUpdate/onMount

```
import { onMount, afterUpdate } from 'svelte';
```

GuessTheSong

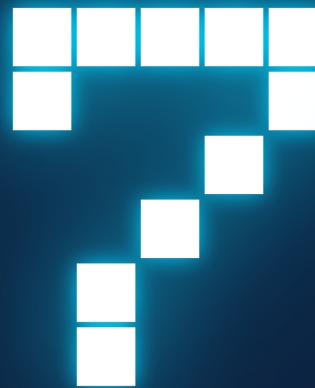
```
onMount(() => {
  selectSongAndAlternatives();
});

afterUpdate(() => {
  selectSongAndAlternatives();
  let audio = document.getElementById('audioPlayer');
  if (audio) {
    audio.src = canzone_da_indovinare['Track Preview URL'];
    audio.load();
  }
});
```

SongCard

```
onMount(() => {
  if(my_songs.some((song) => song['Track URI'] === id)){
    state = true
  } else {
    state = false
  });
};

afterUpdate(() => {
  if(my_songs.some((song) => song['Track URI'] === id)){
    state = true
  } else {
    state = false
  });
});
```



svelte/transition

GuessTheSong

```
import { fade } from 'svelte/transition';

<button transition:fade on:click={() => {
    if (track['Track URI'] === canzone_da_inдовинare['Track URI'])
        alert('Correct!');
        disp('win', "win");
    } else {
        alert('Wrong!');
        disp('lose', "lose");
    }
    selectSongAndAlternatives();
    let audio = document.getElementById('audioPlayer');
    if (audio) {
        audio.src = canzone_da_inдовинare['Track Preview URL'];
        audio.load();
    }
}}>
    {track['Track Name']} - {track['Artist Name(s)']}
</button>
```

Play

```
import { quintOut } from 'svelte/easing';
import { draw } from 'svelte/transition';

<svg
  xmlns="http://www.w3.org/2000/svg"
  fill="none"
  viewBox="0 0 24 24"
  stroke-width="1.5"
  stroke="currentColor"
  class="w-6 h-6">
  <path
    transition:draw={{ duration: 5000, delay: 500, easing: quintOut }}
    stroke-linecap="round"
    stroke-linejoin="round"
    d="M3 3v1.5M3 21v-6m0 0 2.77-.693a9 9 0 0 1 6.208.682l.108.054a9 9 0
  </svg>
```