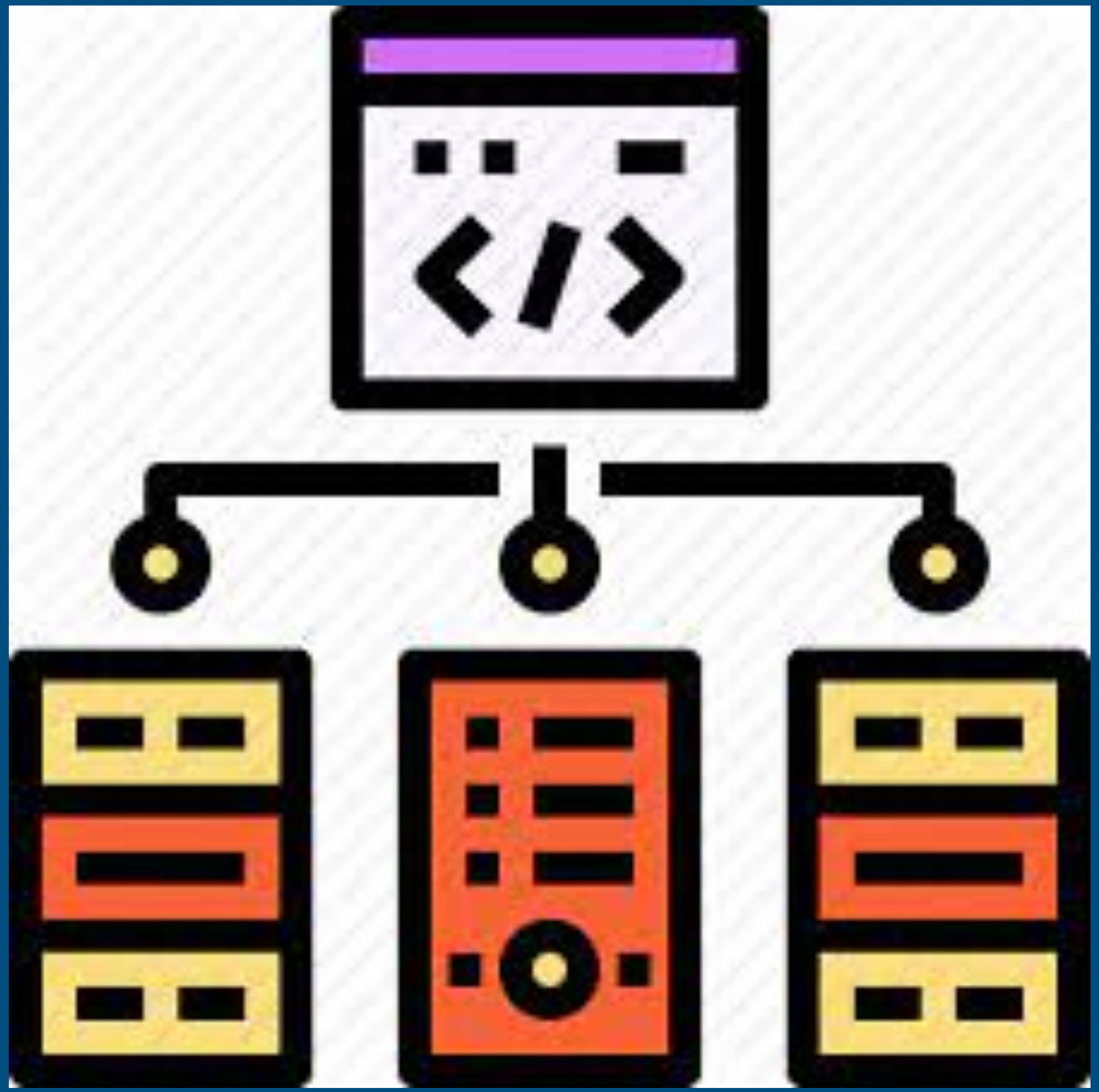
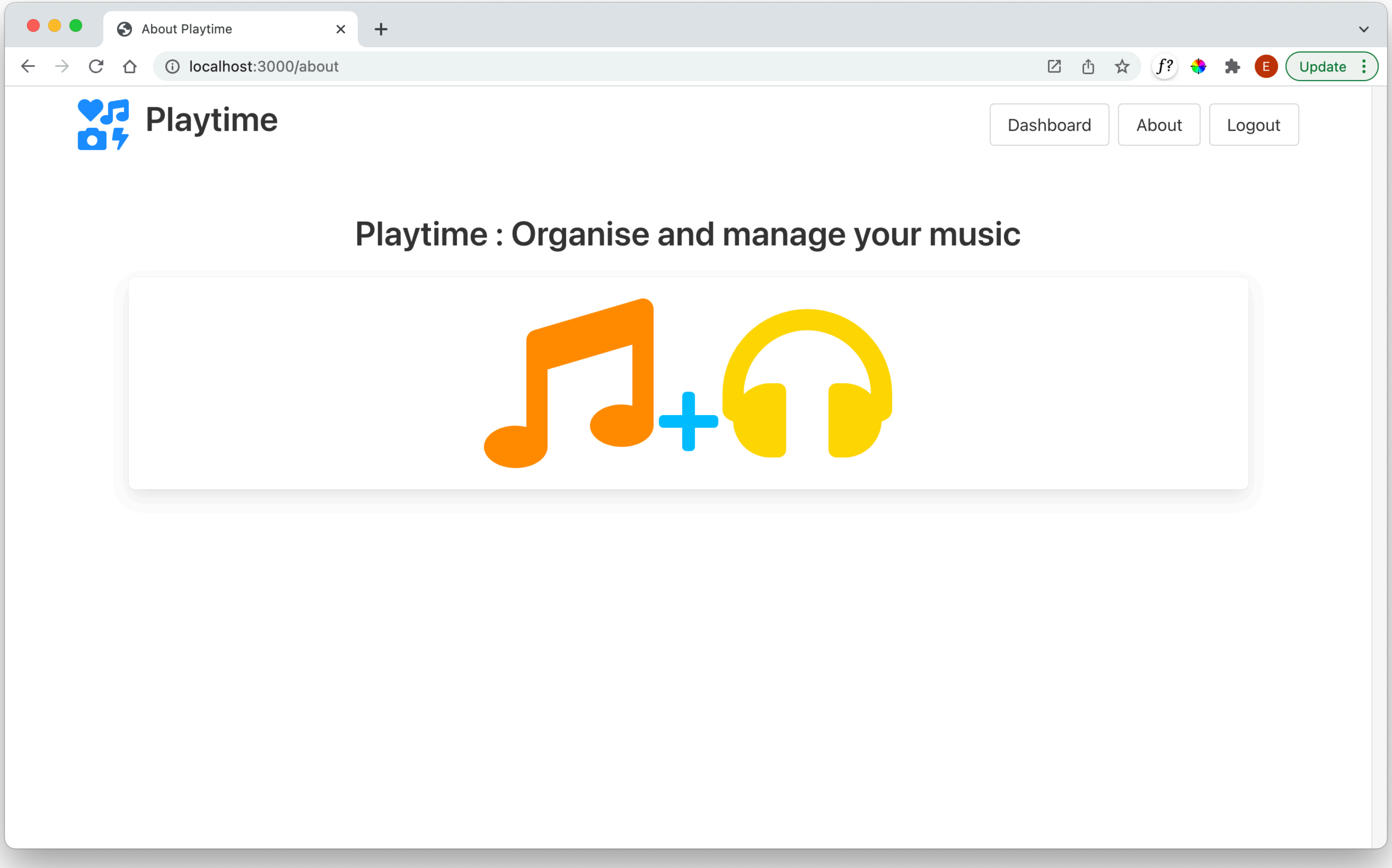
Model View



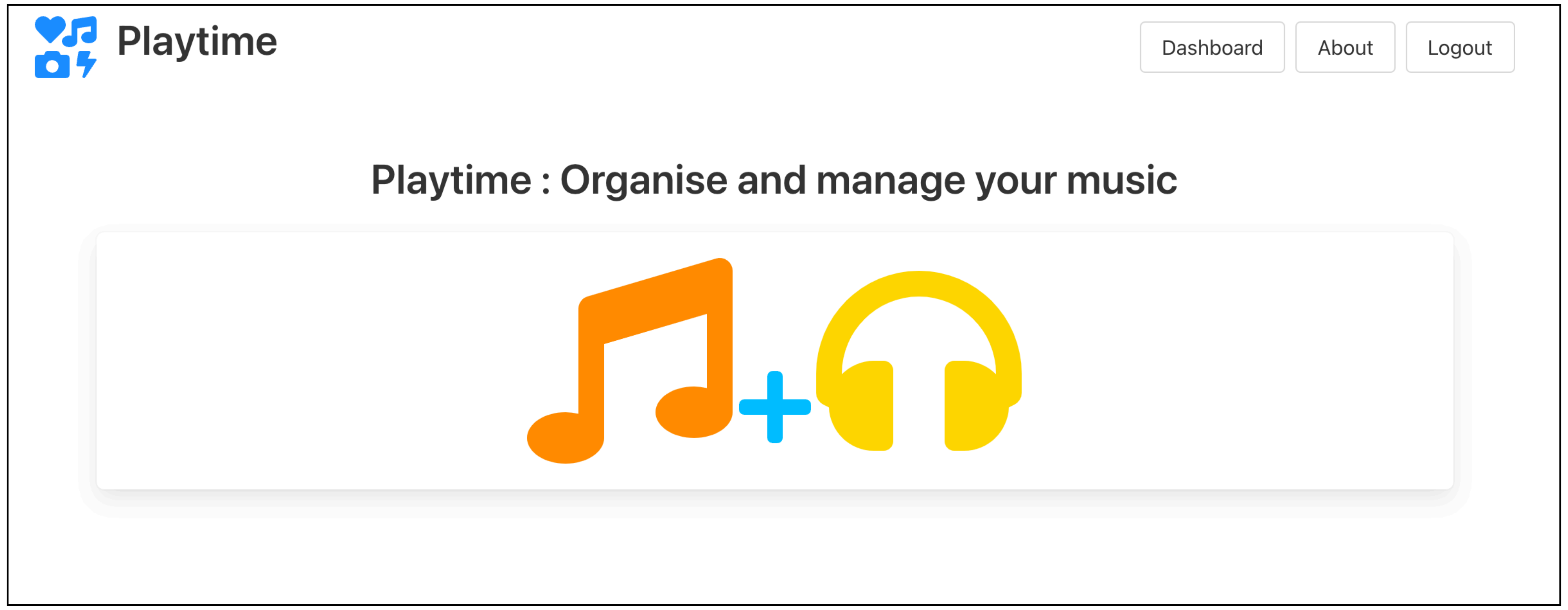
Full Stack Web Development

About View Tracks

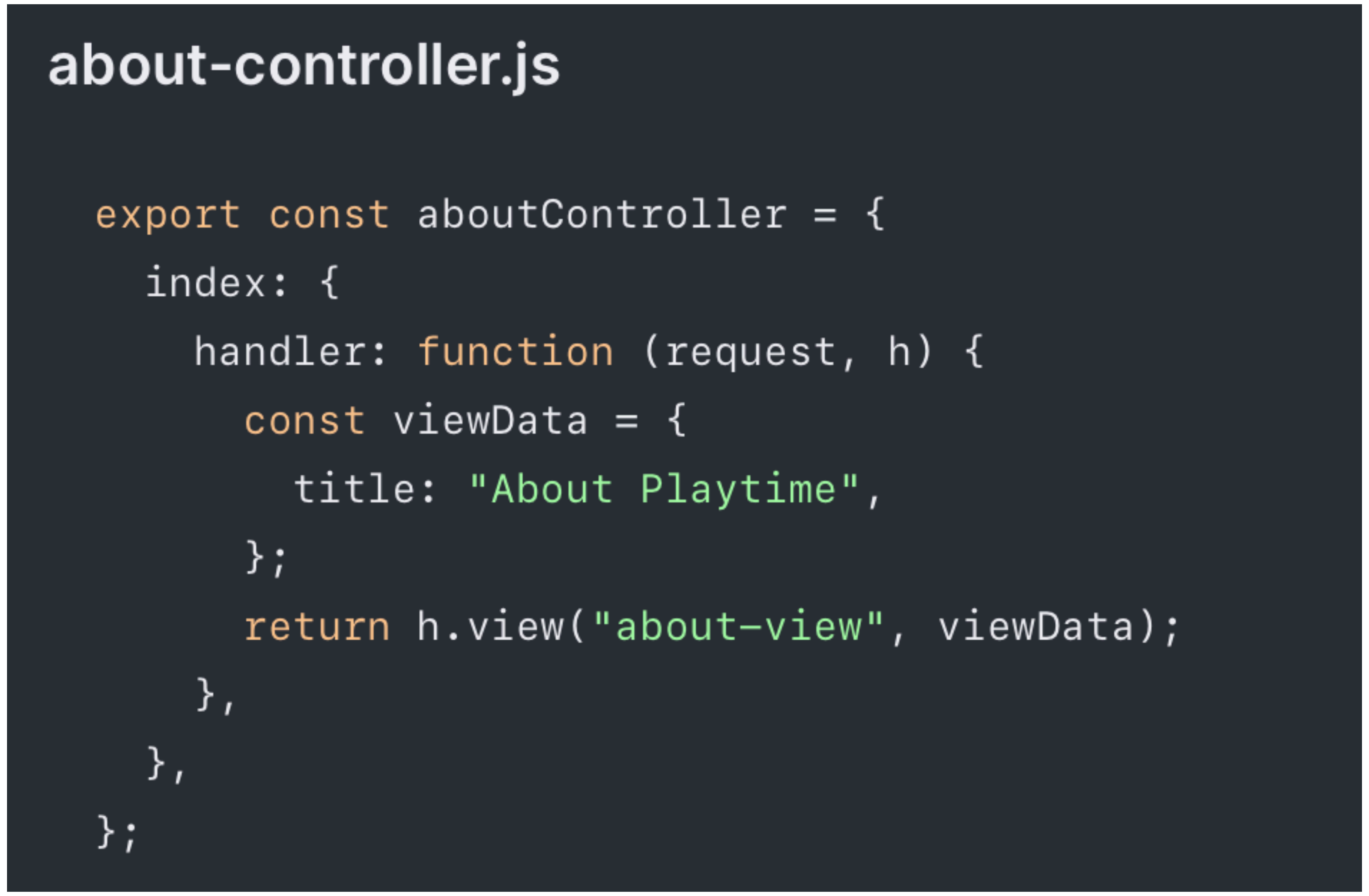
About View



Playtime : Organise and manage your music

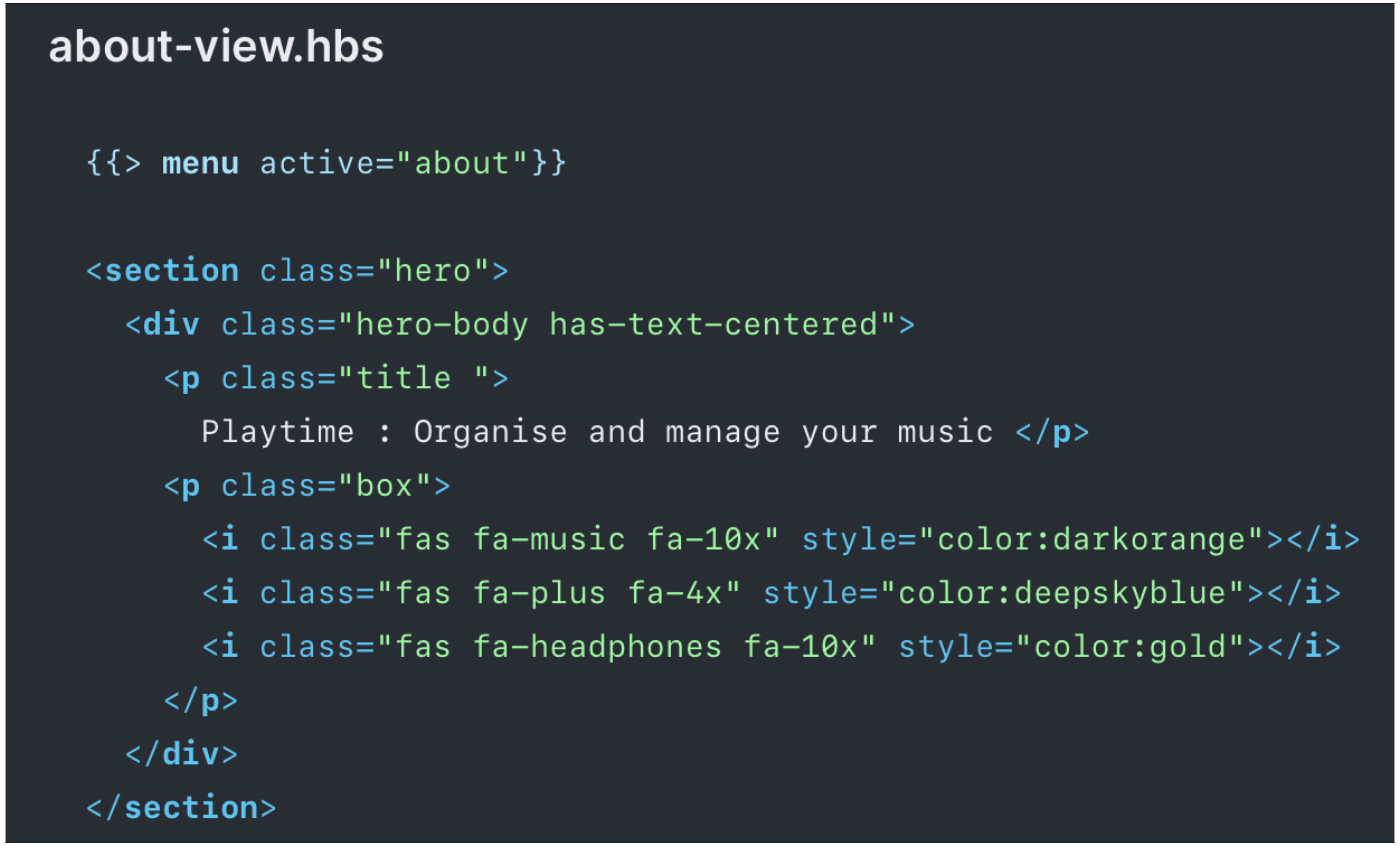


Playtime : Organise and manage your music



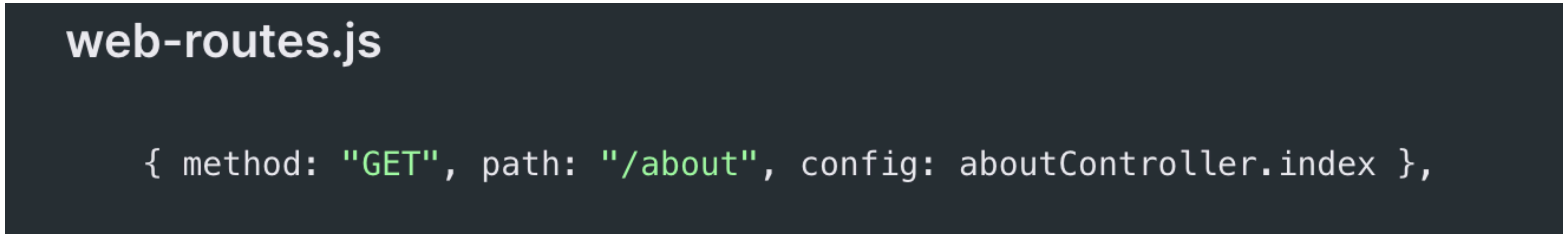
about-controller.js

```
export const aboutController = {
  index: {
    handler: function (request, h) {
      const viewData = {
        title: "About Playtime",
      };
      return h.view("about-view", viewData);
    },
  },
};
```



about-view.hbs

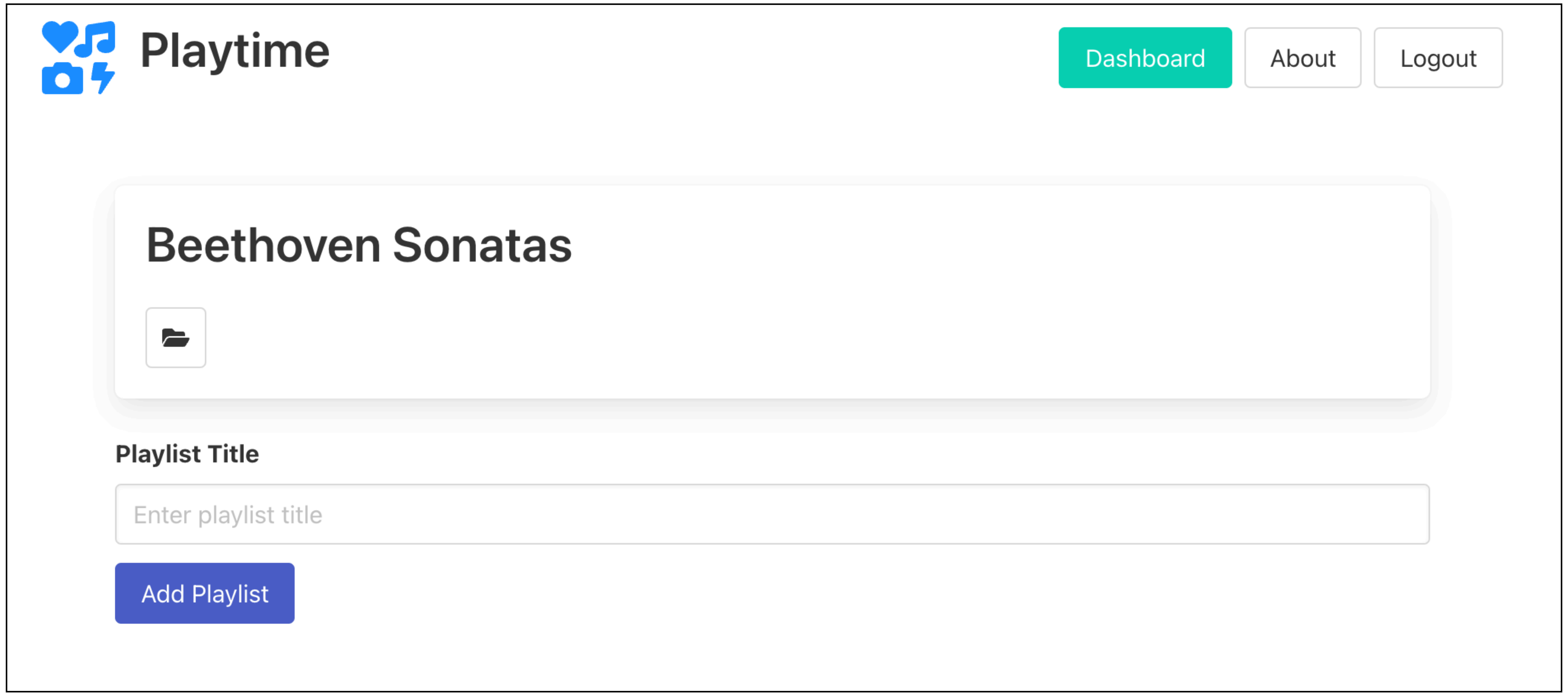
```
{{> menu active="about"}}

<section class="hero">
  <div class="hero-body has-text-centered">
    <p class="title ">
      Playtime : Organise and manage your music </p>
    <p class="box">
      <i class="fas fa-music fa-10x" style="color:darkorange"></i>
      <i class="fas fa-plus fa-4x" style="color:deepskyblue"></i>
      <i class="fas fa-headphones fa-10x" style="color:gold"></i>
    </p>
  </div>
</section>
```



web-routes.js

```
{ method: "GET", path: "/about", config: aboutController.index },
```

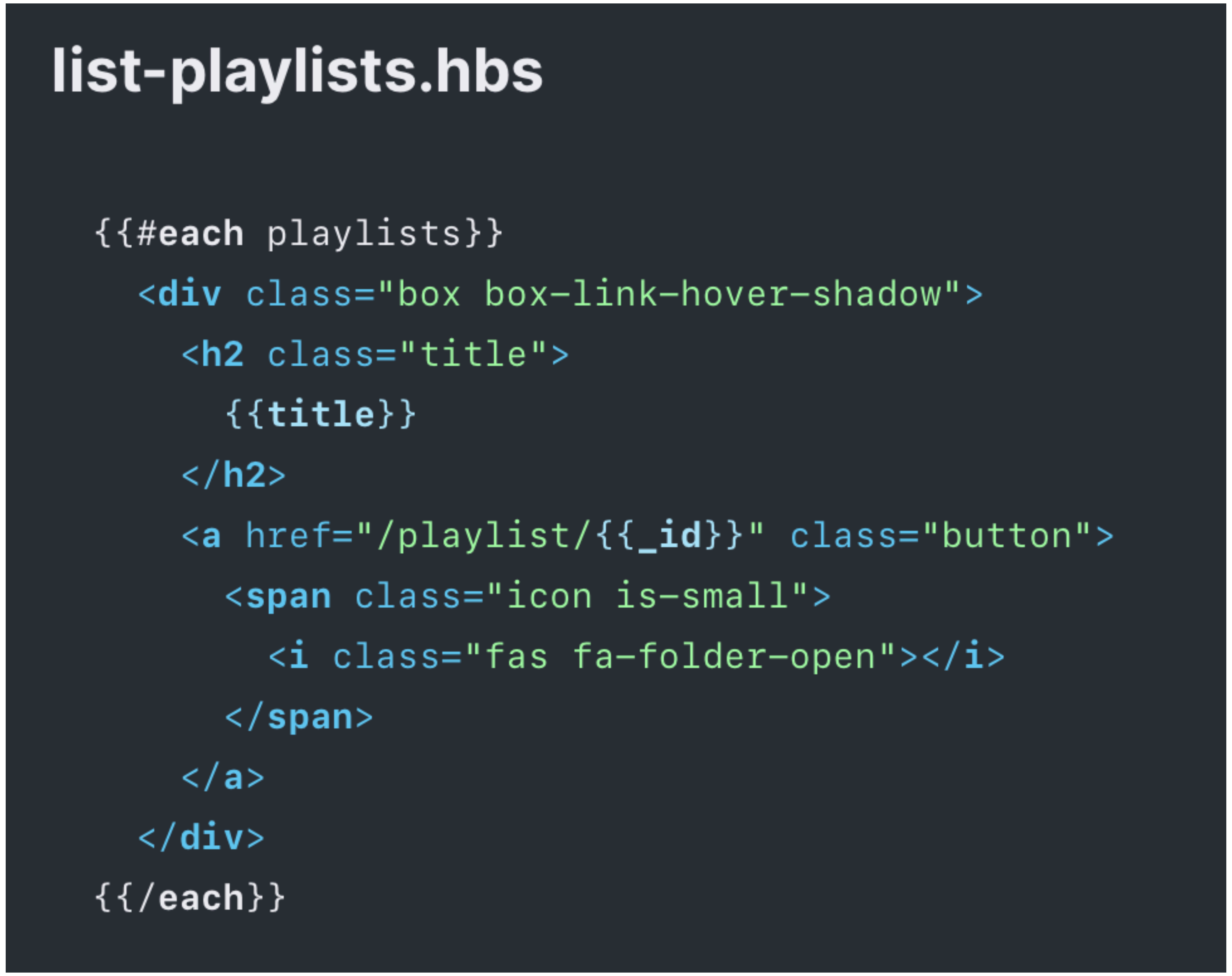
Tracks



Beethoven Sonatas

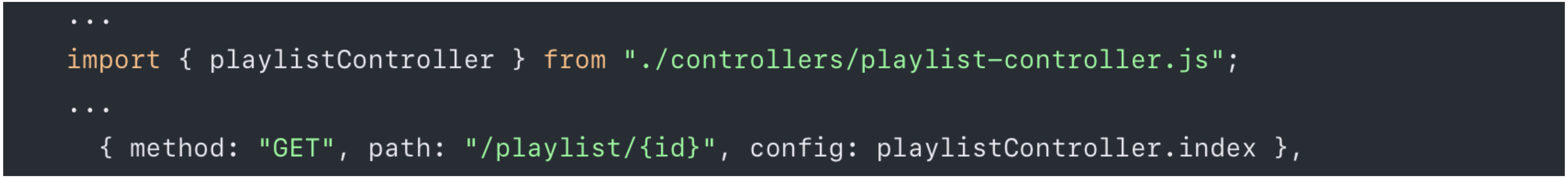
Playlist Title

Enter playlist title

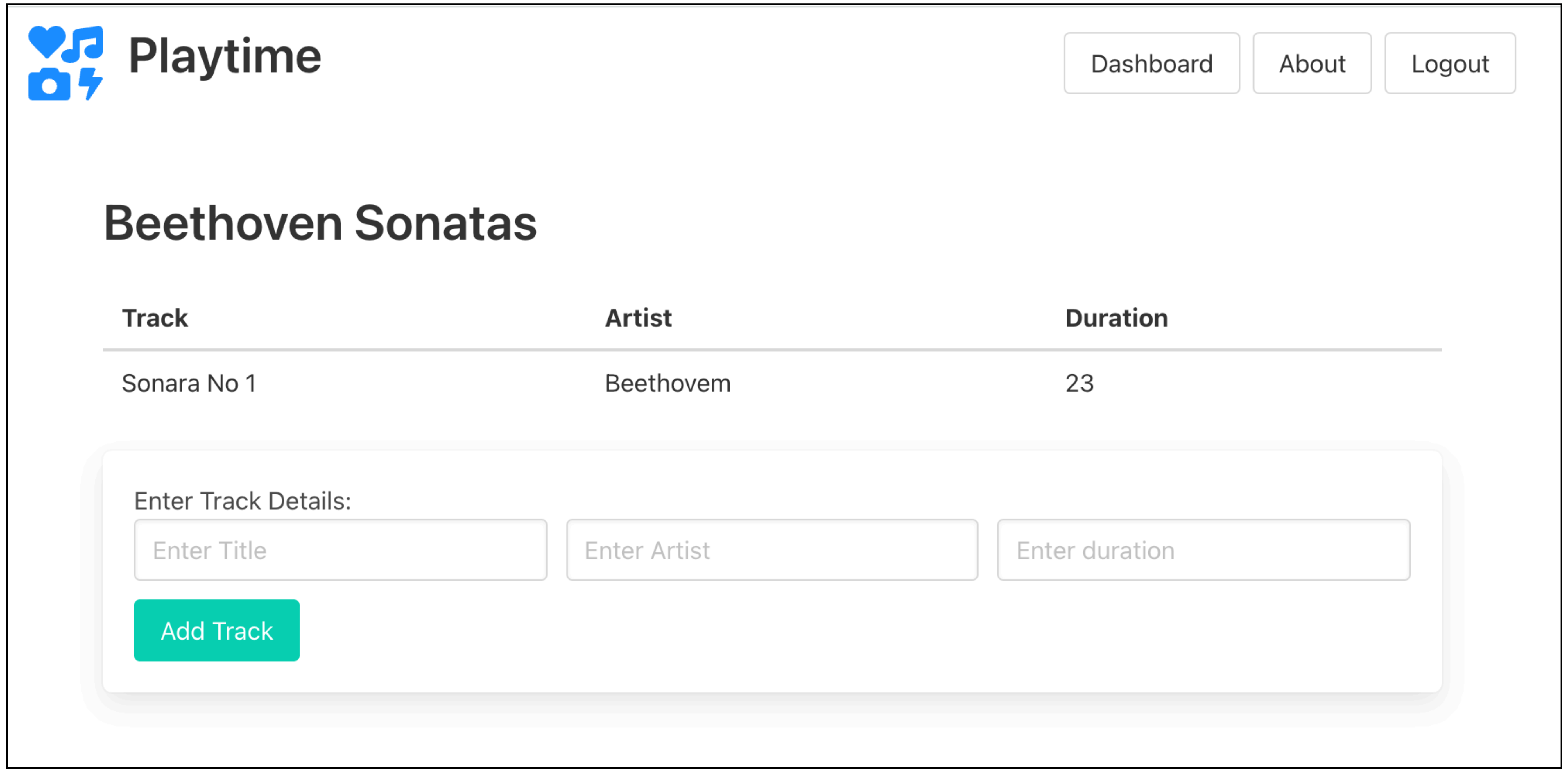


list-playlists.hbs

```
{{#each playlists}}
  <div class="box box-link-hover-shadow">
    <h2 class="title">
      {{title}}
    </h2>
    <a href="/playlist/{{_id}}" class="button">
      <span class="icon is-small">
        <i class="fas fa-folder-open"></i>
      </span>
    </a>
  </div>
{{/each}}
```



```
...
import { playlistController } from "./controllers/playlist-controller.js";
...
  { method: "GET", path: "/playlist/{id}", config: playlistController.index },
```



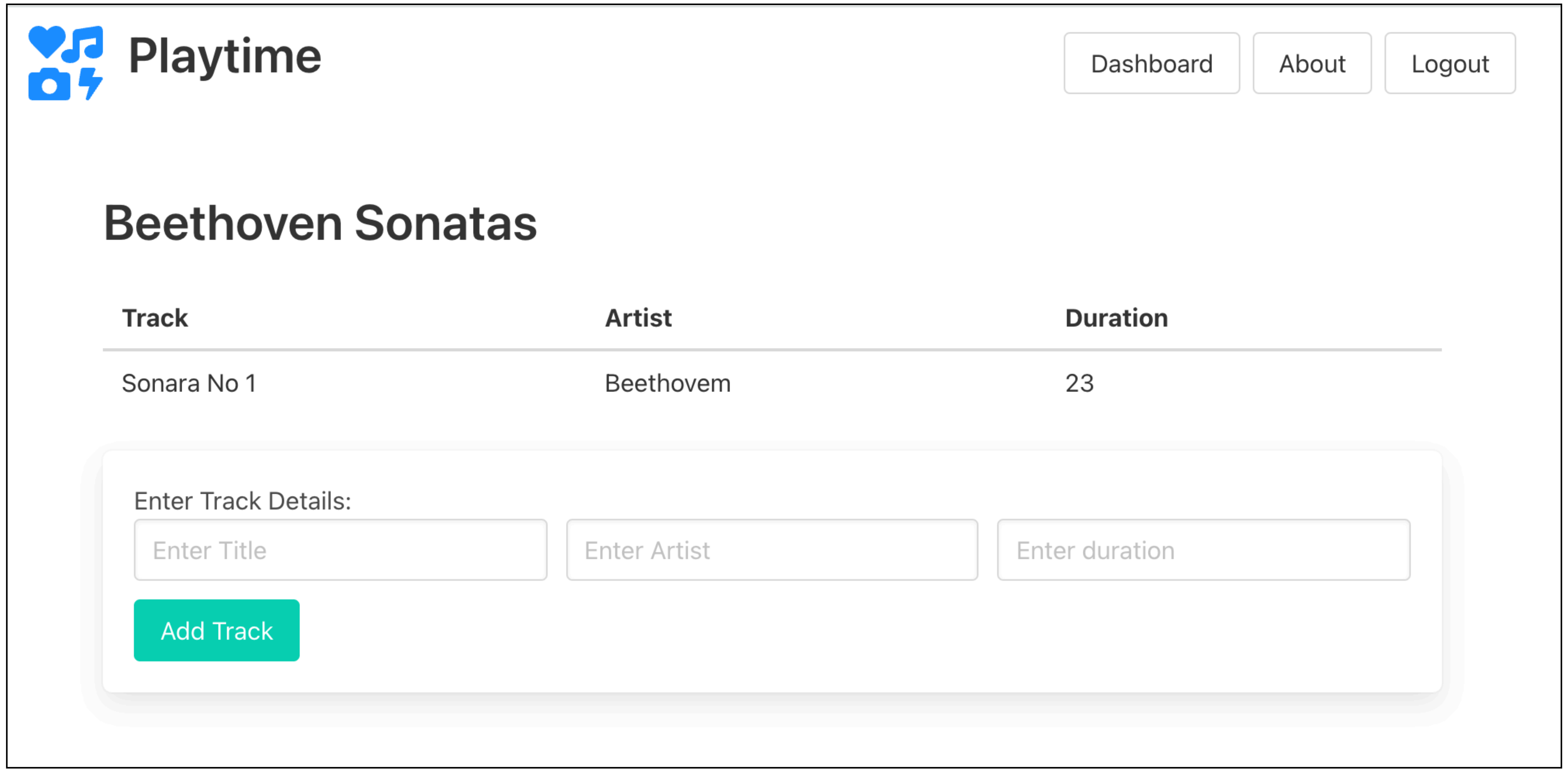
Beethoven Sonatas

Track	Artist	Duration
Sonara No 1	Beethovem	23

Enter Track Details:

list-tracks.hbs

```
<table class="table is-fullwidth">
  <thead>
    <tr>
      <th>Track</th>
      <th>Artist</th>
      <th>Duration</th>
    </tr>
  </thead>
  <tbody>
    {{#each playlist.tracks}}
      <tr>
        <td>
          {{title}}
        </td>
        <td>
          {{artist}}
        </td>
        <td>
          {{duration}}
        </td>
      </tr>
    {{/each}}
  </tbody>
</table>
```



Beethoven Sonatas

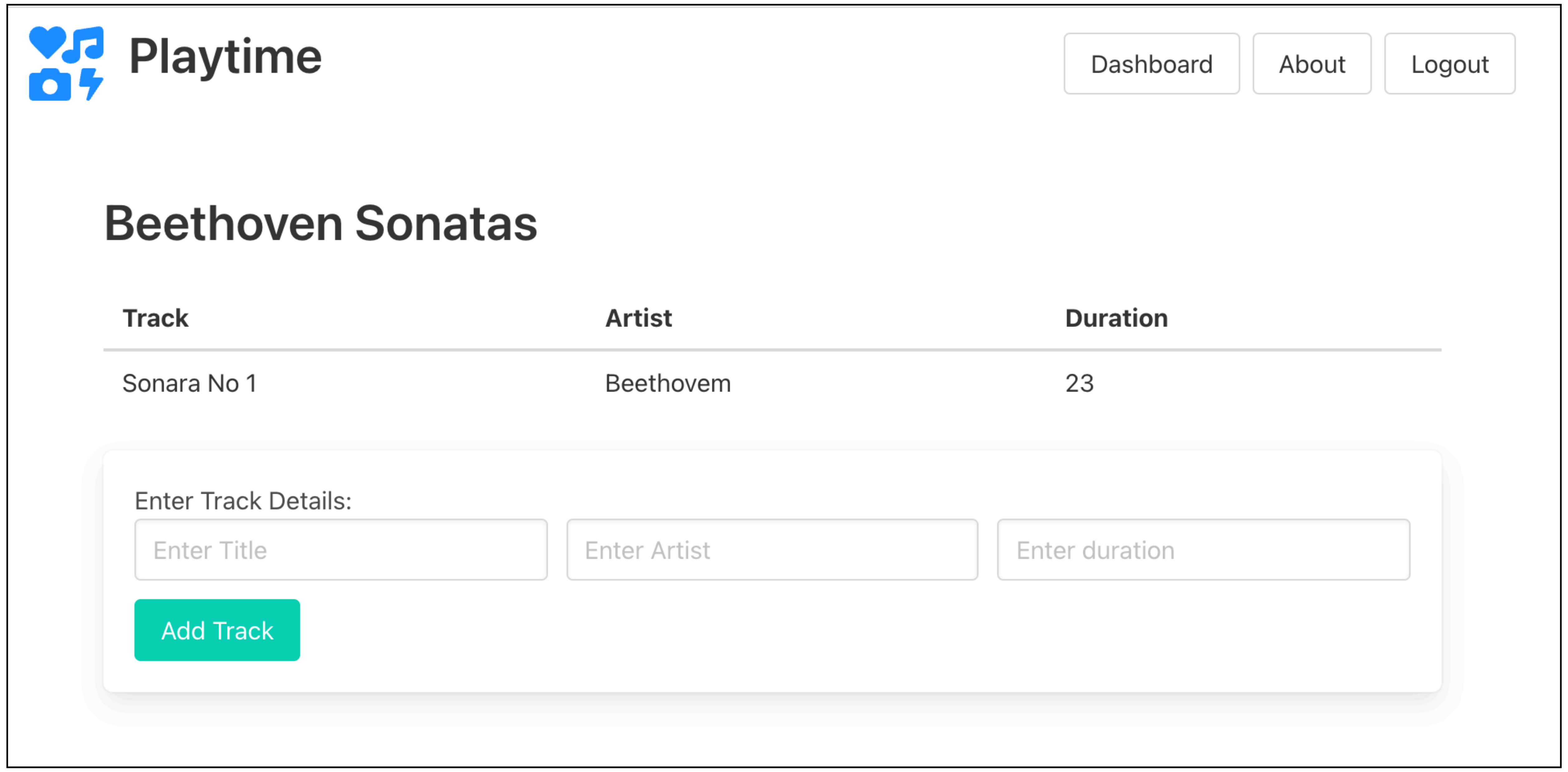
Track	Artist	Duration
Sonara No 1	Beethovem	23

add-tracks.hbs

```
<form class="box" action="/playlist/{{playlist._id}}/addtrack" method="POST">
  <label>Enter Track Details:</label>
  <div class="field is-horizontal">
    <div class="field-body">
      <div class="field">
        <input class="input" type="text" placeholder="Enter Title" name="title">
      </div>
      <div class="field">
        <input class="input" type="text" placeholder="Enter Artist" name="artist">
      </div>
      <div class="field">
        <input class="input" type="text" placeholder="Enter duration" name="duration">
      </div>
    </div>
  </div>
  <button class="button is-primary">Add Track</button>
</form>
```

playlist-view.hbs

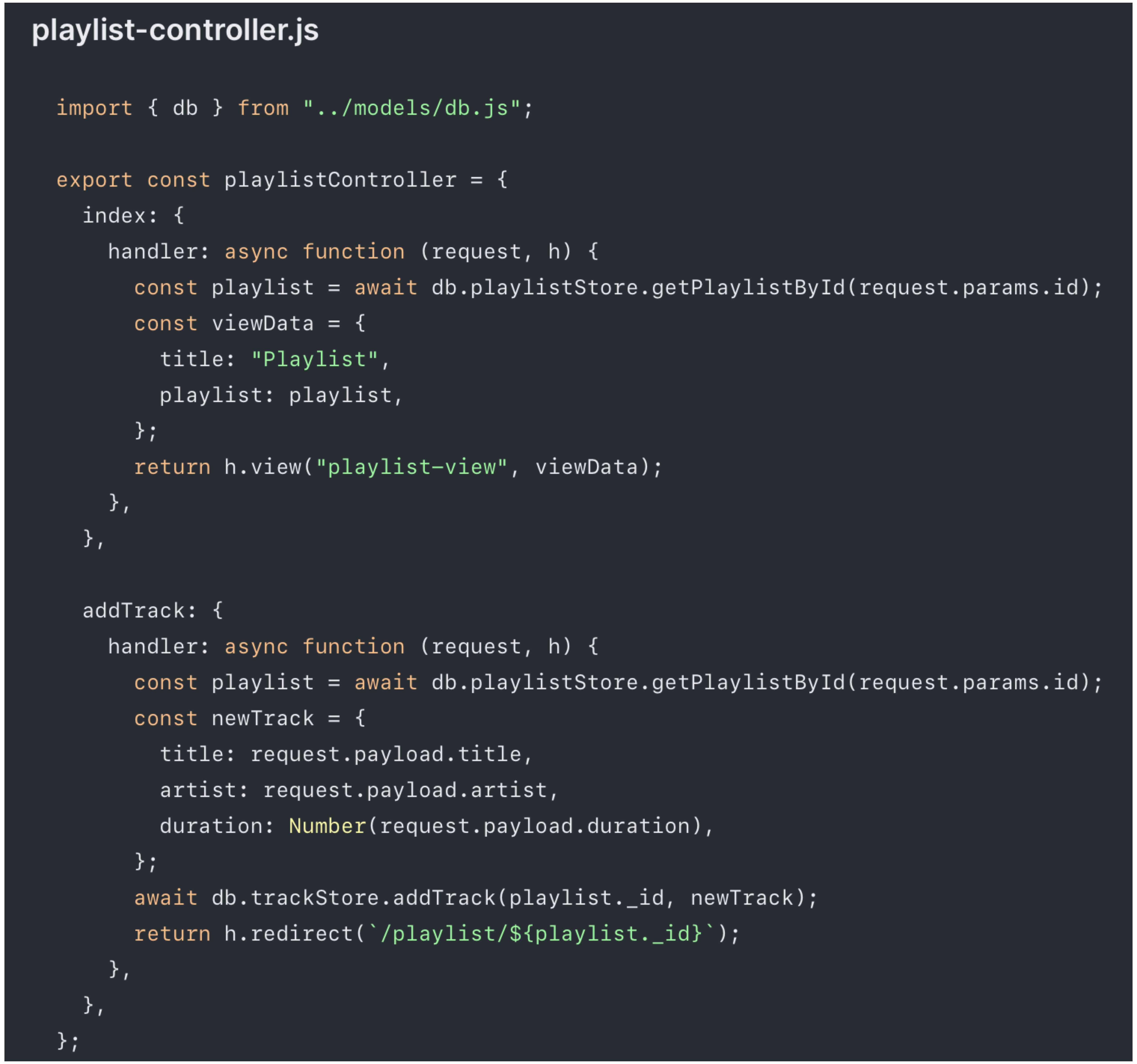
```
{{> menu}}

<section class="section">
  <div class="title">
    {{playlist.title}}
  </div>
  {{> list-tracks}}
  {{> add-track}}
</section>
```



Beethoven Sonatas

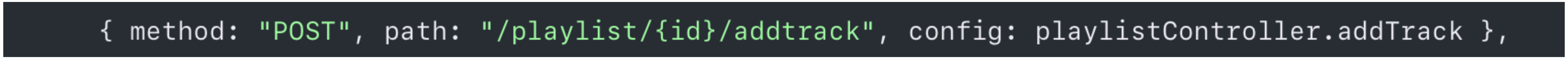
Track	Artist	Duration
Sonara No 1	Beethovem	23

Enter Track Details:

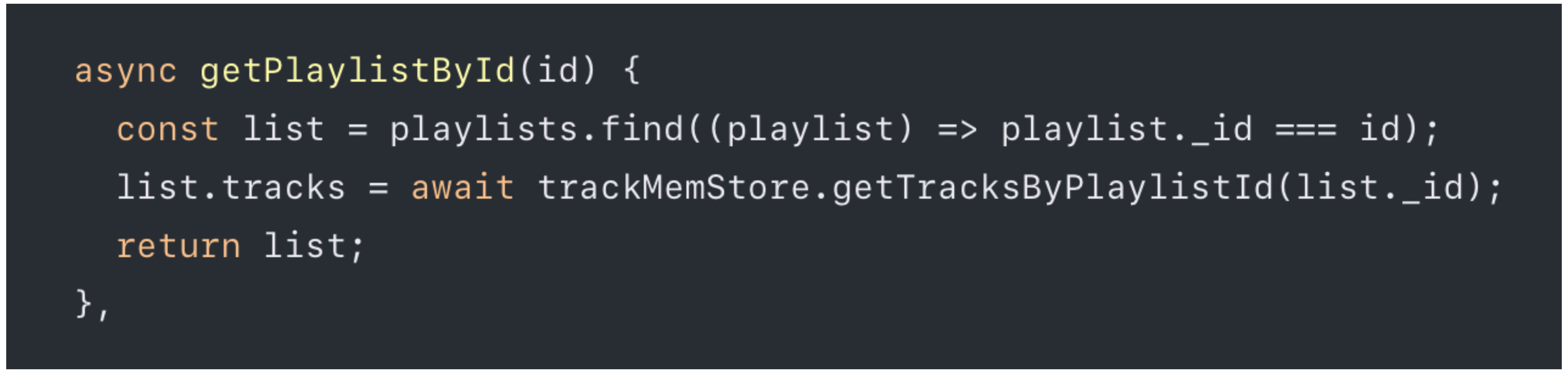


playlist-controller.js

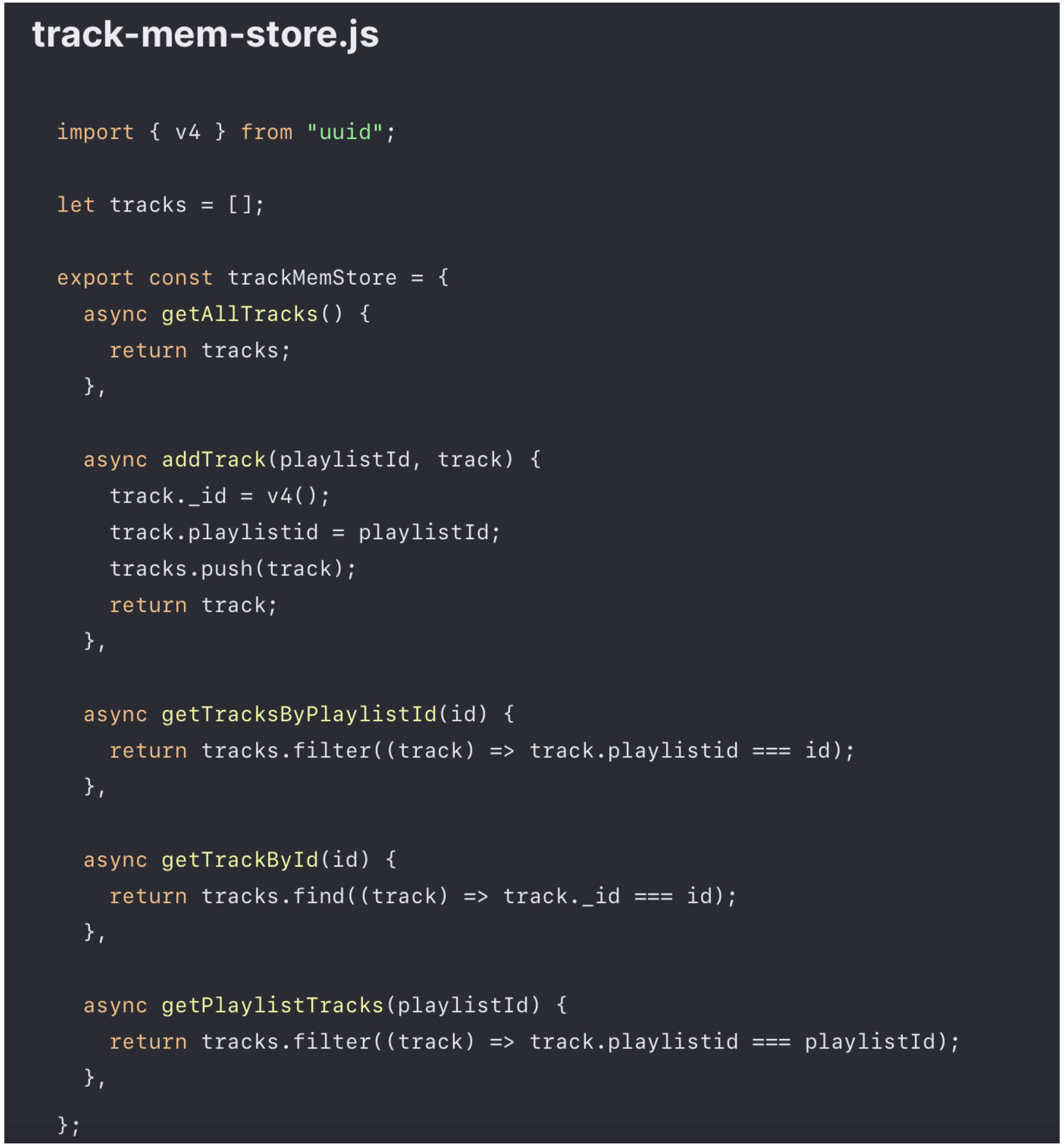
```
import { db } from "../models/db.js";

export const playlistController = {
  index: {
    handler: async function (request, h) {
      const playlist = await db.playlistStore.getPlaylistById(request.params.id);
      const viewData = {
        title: "Playlist",
        playlist: playlist,
      };
      return h.view("playlist-view", viewData);
    },
  },

  addTrack: {
    handler: async function (request, h) {
      const playlist = await db.playlistStore.getPlaylistById(request.params.id);
      const newTrack = {
        title: request.payload.title,
        artist: request.payload.artist,
        duration: Number(request.payload.duration),
      };
      await db.trackStore.addTrack(playlist._id, newTrack);
      return h.redirect(`/playlist/${playlist._id}`);
    },
  },
};
```



```
{ method: "POST", path: "/playlist/{id}/addtrack", config: playlistController.addTrack },
```



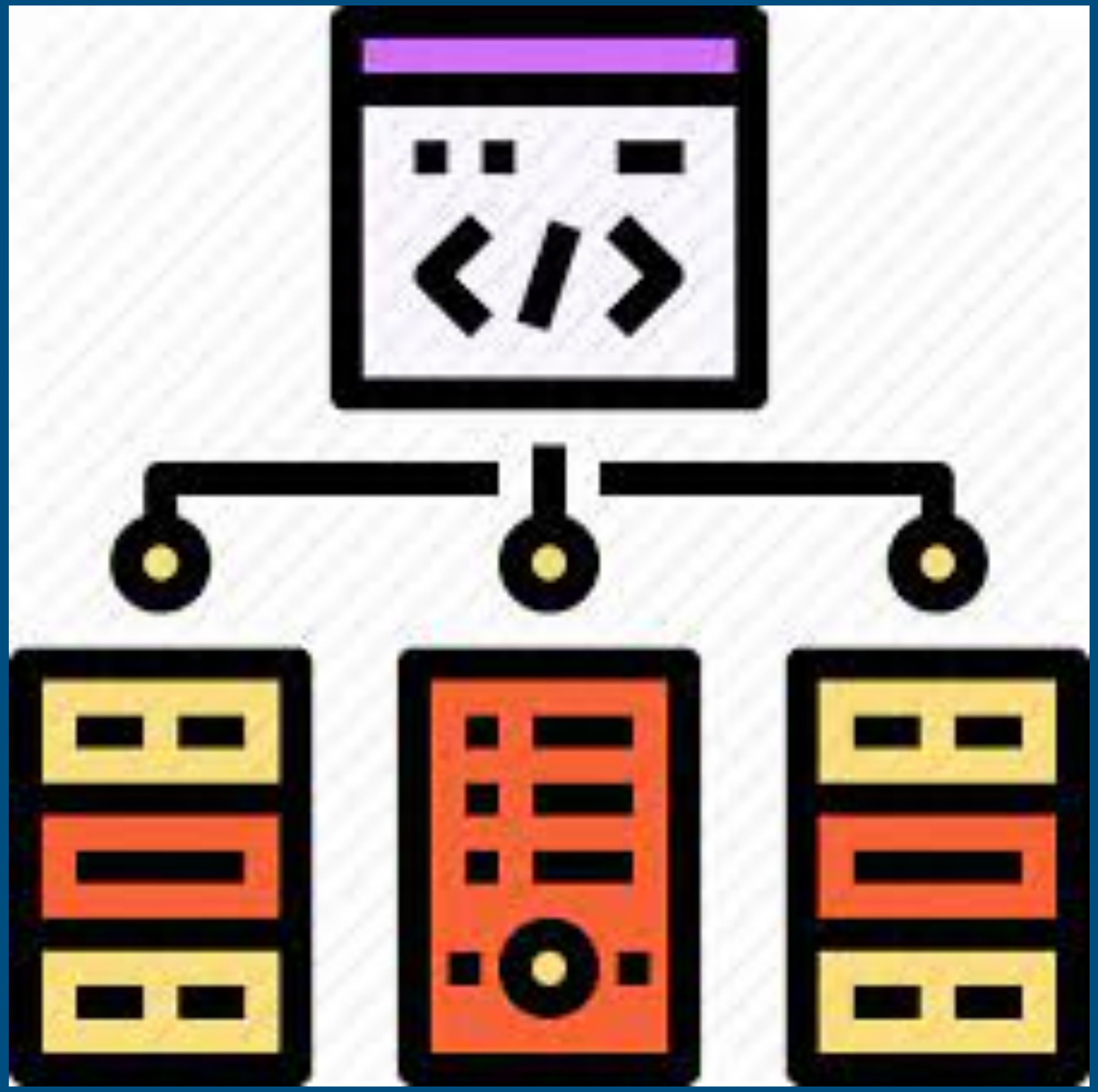
```
async getPlaylistById(id) {
  const list = playlists.find((playlist) => playlist._id === id);
  list.tracks = await trackMemStore.getTracksByPlaylistId(list._id);
  return list;
},
```



track-mem-store.js

```
import { v4 } from "uuid";

let tracks = [];

export const trackMemStore = {
  async getAllTracks() {
    return tracks;
  },

  async addTrack(playlistId, track) {
    track._id = v4();
    track.playlistid = playlistId;
    tracks.push(track);
    return track;
  },

  async getTracksByPlaylistId(id) {
    return tracks.filter((track) => track.playlistid === id);
  },

  async getTrackById(id) {
    return tracks.find((track) => track._id === id);
  },

  async getPlaylistTracks(playlistId) {
    return tracks.filter((track) => track.playlistid === playlistId);
  },
};
```

Model View



Full Stack Web Development