

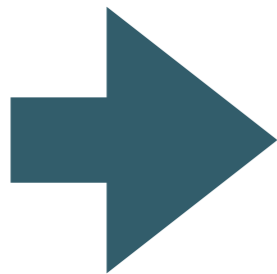
# Playlist Controller

---

Playlist 1		Dashboard	About
<b>Beethoven Sonatas</b>			
Song	Artist		
Piano Sonata No. 3	Beethoven		
Piano Sonata No. 7	Beethoven		
Piano Sonata No. 10	Beethoven		
<b>Beethoven Concertos</b>			
Song	Artist		
Piano Concerto No. 0			
Piano Concerto No. 4			
Piano Concerto No. 6			

# Revised Dashboard

Playlist 1	Dashboard	About
<b>Beethoven Sonatas</b> Total Duration: 35 <a href="#">View</a>		
<b>Beethoven Concertos</b> Total Duration: 23 <a href="#">View</a>		
<b>Beethoven Variations</b> Total Duration: 67 <a href="#">View</a>		



- Shows summary list of playlists

## Beethoven Sonatas

Total Duration: 35

[View](#)

## Beethoven Concertos

Total

[View](#)

## Beethoven

Total

[View](#)

### Beethoven Sonatas

Song	Artist
Piano Sonata No. 3	Beethoven
Piano Sonata No. 7	Beethoven
Piano Sonata No. 10	Beethoven

- Full playlist displayed when 'View' selected

# Dashboard.hbs

```
{{> menu id="dashboard"}}

{{#each playlists}}
  <section class="ui segment">
    <h2 class="ui header">
      {{title}}
    </h2>
    <table class="ui table">
      <thead>
        <tr>
          <th>Song</th>
          <th>Artist</th>
        </tr>
      </thead>
      <tbody>
        {{#each songs}}
          <tr>
            <td>
              {{title}}
            </td>
            <td>
              {{artist}}
            </td>
          </tr>
        {{/each}}
      </tbody>
    </table>
  </section>
{{/each}}
```

No change to controller

```
const dashboard = {
  index(request, response) {
    logger.info('dashboard rendering');
    const viewData = {
      title: 'Playlist Dashboard',
      playlists: playlistCollection,
    };
    logger.info('about to render', playlistCollection);
    response.render('dashboard', viewData);
  },
};
```

```
{{> menu id="dashboard"}}

{{#each playlists}}
  <section class="ui segment">
    <h2 class="ui header">
      {{title}}
    </h2>
    <p> Total Duration: {{duration}} </p>
    <a href="/playlist/{{id}}"> View </a>
  </section>
{{/each}}
```

Revised dashboard.hbs

```
{{> menu id="dashboard"}}  
  
{{#each playlists}}  
  <section class="ui segment">  
    <h2 class="ui header">  
      {{title}}  
    </h2>  
    <p> Total Duration: {{duration}} </p>  
    <a href="/playlist/{{id}}"> View </a>  
  </section>  
{{/each}}
```

## Dashboard View

New Route

## routes.js

- New Route not initially supported

```
router.get('/', start.index);  
router.get('/dashboard', dashboard.index);  
router.get('/about', about.index);
```

- Error message when link pressed

Secure | https://iot.gomix.me/playlist/01

Cannot GET /playlist/01

# Introducing a new view

---

- Typically need 3 things:
  - Controller
  - View
  - Route

controller:  
playlist.js

```
'use strict';

const logger = require('../utils/logger');
const playlistCollection = require('../models/playlist-store.js');

const playlist = {
  index(request, response) {
    const viewData = {
      title: 'Playlist',
    };
    response.render('playlist', viewData);
  },
};

module.exports = playlist;
```

view:  
playlist.hbs

```
{{> menu}}

<section class="ui center aligned middle aligned segment">
  <h2 class="ui header">
    {{playlist.title}}
  </h2>
</section>
```

route:  
routes.js

```
router.get('/playlist/:id', playlist.index);
```

# Controller

---

- In order to display the correct playlist, we need to extract the id from the url.



```
index(request, response) {  
  const playlistId = request.params.id;  
  logger.debug('Playlist id = ', playlistId);  
  const viewData = {  
    title: 'Playlist',  
  };  
  response.render('playlist', viewData);  
}
```



# Revised Model

---

- playlistStore is an object with 2 methods:
  - retrieve all playlists from store
  - retrieve a specific playlist by its id from the store.

```
const playlistStore = {  
  playlistCollection: require('./playlist-store.json').playlistCollection,  
  getAllPlaylists() {  
    return this.playlistCollection;  
  },  
  getPlaylist(id) {  
    return _.find(this.playlistCollection, { id: id });  
  },  
};
```

*Don't worry about how this is implemented for the moment!*

# Revised playlist Controller

---

- Recover playlist ID is request
- Get playlist matching this id and insert into viewData
- Pass **viewData** to to the view (containing the playlist)

```
const playlistStore = require('../models/playlist-store');

const playlist = {
  index(request, response) {

    const playlistId = request.params.id;
    logger.debug('Playlist id = ', playlistId);

    const viewData = {
      title: 'Playlist',
      playlist: playlistStore.getPlaylist(playlistId),
    };

    response.render('playlist', viewData);
  },
};
```

# Revised Playlist View

```
index(request, response) {  
  
  const playlistId = request.params.id;  
  logger.debug('Playlist id = ', playlistId);  
  
  const viewData = {  
    title: 'Playlist',  
    playlist: playlistStore.getPlaylist(playlistId),  
  };  
  
  response.render('playlist', viewData);  
}
```

Beethoven Sonatas	
Song	Artist
Piano Sonata No. 3	Beethoven
Piano Sonata No. 7	Beethoven
Piano Sonata No. 10	Beethoven

```
<table class="ui table">  
  <thead>  
    <tr>  
      <th>Song</th>  
      <th>Artist</th>  
    </tr>  
  </thead>  
  <tbody>  
    {{#each playlist.songs}}  
      <tr>  
        <td>  
          {{title}}  
        </td>  
        <td>  
          {{artist}}  
        </td>  
      </tr>  
    {{/each}}  
  </tbody>  
</table>
```

listsongs.hbs

```
{{> menu}}  
  
<section class="ui center aligned middle aligned segment">  
  <h2 class="ui header">  
    {{playlist.title}}  
  </h2>  
  {{> listsongs}}  
</section>
```

playlist.hbs