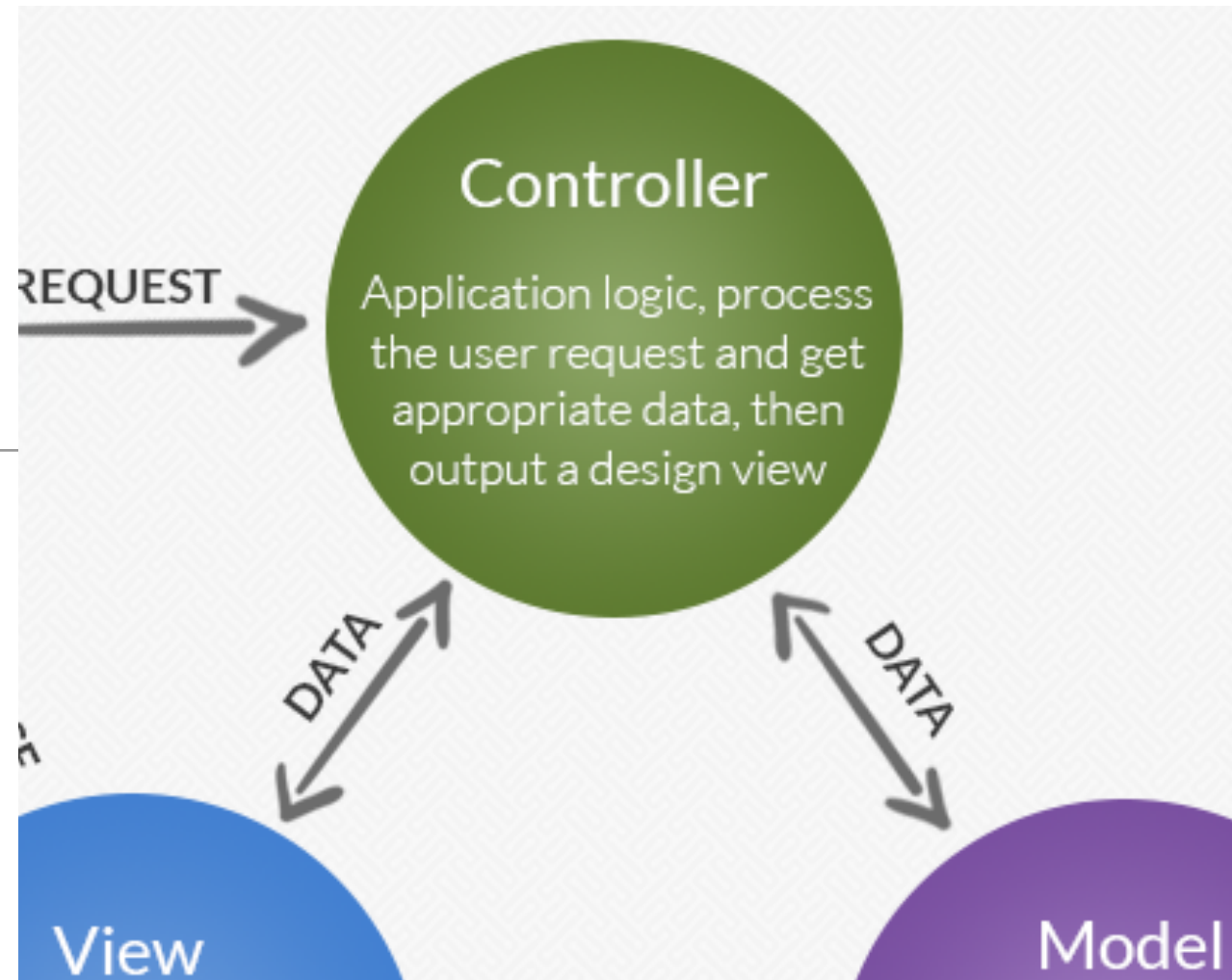


# Controllers

---



# Feature

## Beethoven Sonatas

Total Duration: 35

[View](#)

## Beethoven Concertos

Total Duration: 23

[View](#)

## Beethoven Variations

Total Duration: 67

[View](#)

Open playlists in  
separate view...

## Beethoven Sonatas

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

# Dashboard

Retrieve all playlists  
Send to dashboard

```
public class Dashboard extends Controller
{
    public static void index()
    {
        Logger.info("Rendering Admin");

        List<Playlist> playlists = Playlist.findAll();
        render ("dashboard.html", playlists);
    }
}
```

## Beethoven Sonatas

Total Duration: 35

[View](#)

## Beethoven Concertos

Total Duration: 23

[View](#)

## Beethoven Variations

Total Duration: 67

[View](#)

```
{% extends 'main.html' %}
{% set title = 'Dashboard' %}

{% menu id = "dashboard" %}

{% list items = playlists, as = 'playlist' %}
    <section class="ui segment">
        <h2 class="ui header">
            {{ playlist.title }}
        </h2>
        <p> Total Duration: {{ playlist.duration }} </p>
        <a href="/playlists/{{ playlist.id }}"> View </a>
    </section>
{% /list %}
```

# /playlists route Not Found

## Beethoven Variations

Total Duration: 67

View

```
#{extends 'main.html' /}  
#{set title:'Dashboard' /}  
  
#{menu id:"dashboard"/}  
  
#{list items:playlists, as:'playlist'}  
  <section class="ui segment">  
    <h2 class="ui header">  
      ${playlist.title}  
    </h2>  
    <p> Total Duration: ${playlist.duration} </p>  
    <a href="/playlists/${playlist.id}"> View </a>  
  </section>  
#{/list}
```

localhost:9000/playlist/2

## Not found

playlist.1 action not found

These routes have been tried, in this order :

1. GET	/@projectdocs/files/{name}	ProjectDocumentation.file
2. GET	/@projectdocs/images/{name}	ProjectDocumentation.image
3. GET	/@projectdocs/{id}	ProjectDocumentation.page
4. GET	/@projectdocs/?	ProjectDocumentation.index
5. GET	/@documentation/{docLang}/cheatsheet/{category}	PlayDocumentation.cheatSheet
6. GET	/@documentation/cheatsheet/{category}	PlayDocumentation.cheatSheet
7. GET	/@documentation/modules/{module}/files/{name}	PlayDocumentation.file
8. GET	/@documentation/modules/{module}/images/{name}	PlayDocumentation.image
9. GET	/@documentation/{docLang}/releases/{version}/{id}	PlayDocumentation.releases
10. GET	/@documentation/{docLang}/releases/{id}	PlayDocumentation.releases
11. GET	/@documentation/{docLang}/modules/{module}/{id}	PlayDocumentation.page
12. GET	/@documentation/{docLang}/files/{name}	PlayDocumentation.file
13. GET	/@documentation/{docLang}/images/{name}	PlayDocumentation.image
14. GET	/@documentation/files/{name}	PlayDocumentation.file
15. GET	/@documentation/images/{name}	PlayDocumentation.image

# PlaylistCtrl Controller

GET      /playlists/{id}      PlaylistCtrl.index

conf/routes

localhost:9000/playlist/2

id of playlist

```
public class PlaylistCtrl extends Controller
{
    public static void index()
    {
        render("playlist.html");
    }
}
```

PlaylistCtrl.java

```
{extends 'main.html' /}
#{set title:'Playlist' /}

#{menu id:"dashboard"/}

<section class="ui segment">
    <h2 class="ui header">
        Playlist details...
    </h2>
</section>
```

playlist.html

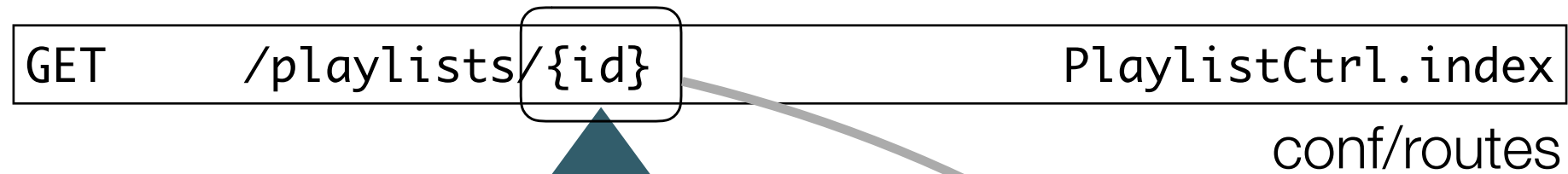
Playlist

Dashboard

About

Playlist details...

# PlaylistCtrl Controller



id of playlist

Find Playlist with  
selected ID

```
public class PlaylistCtrl extends Controller
{
    public static void index(Long id)
    {
        Playlist playlist = Playlist.findById(id);
        Logger.info ("Playlist id = " + id);
        render("playlist.html", playlist);
    }
}
```

Send Playlist to  
View

# Playlist Views

playlist.html

```
{extends 'main.html' /}
{set title:'Playlist' /}

{menu id:"dashboard"/}

<section class="ui segment">
  <h2 class="ui header">
    ${playlist.title}
  </h2>
  {listsongs playlist:playlist /}
</section>
```

listsongs.html

```
<table class="ui fixed table">
  <thead>
    <tr>
      <th>Song</th>
      <th>Artist</th>
      <th>Duration</th>
    </tr>
  </thead>
  <tbody>
    {list items:_playlist.songs, as:'song'}
    <tr>
      <td>
        ${song.title}
      </td>
      <td>
        ${song.artist}
      </td>
      <td>
        ${song.duration}
      </td>
    </tr>
  </tbody>
  {/list}
</tbody>
</table>
```

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

# Delete Song Feature

## Beethoven Sonatas

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



# Delete Song Button

## Beethoven Sonatas

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

```
<td>  
  <a href="/playlists/${_playlist.id}/deletesong/${song.id}" class="ui tiny red button"> Delete Song </a>  
</td>
```

# Delete Song Route

```
<td>  
  <a href= "/playlists/${_playlist.id}/deletesong/${song.id}" class="ui tiny red button"> Delete Song </a>  
</td>
```

listsongs.html

GET

/playlists/{id}/deletesong/{songid}

PlaylistCtrl.deleteSong

conf/routes

```
public static void deletesong (Long id,      Long songid)  
{  
  Playlist playlist = Playlist.findById(id);  
  Song song = Song.findById(songid);  
  Logger.info ("Removing" + song.title);  
  playlist.songs.remove(song);  
  playlist.save();  
  song.delete();  
  render("playlist.html", playlist);  
}
```

PlaylistCtrl.java

# Delete Song Action

Find the Playlist  
Find the song  
Remove the song from the  
playlist collection  
Save the collection  
remove the song from the  
database  
Display the revised playlist

```
public static void deletesong (Long id,      Long songid)
{
    Playlist playlist = Playlist.findById(id);

    Song song = Song.findById(songid);

    playlist.songs.remove(song);
    playlist.save();

    song.delete();

    render("playlist.html", playlist);
}
```

PlaylistCtrl.java