**ACTION MAP – soundcloud**

**CONTROLLERS**

1. **UserController – 12**

**POST – 5**

@/login (body – LoginDTO)

@/logout (user from session)

@/users (body – RegisterDTO)

@/followuser/{user-ID} (user from session)

@/filter-users (body -FilterUserRequestDTO) -> List<UserResponseDTO> results

**DELETE – 2**

@/remove-user (user from session)

@/unfollow-user/{user-ID} (user from session)

**GET – 3**

@/my-profile (user from session)

* // Get all user information for the currently logged user

@/users/{username}

* // Get specific user and public information for their profile

@/users/{username}/download-picture

**PUT – 2**

@/users (body – UpdateDTO)

* Update user personal information

1. **SongController - 8**

**POST – 1**

@/add-song (body – SongDTO, user from session)

@/filter-songs (body – FilterSongRequestDTO) -> List<SongResponseDTO> results

**DELETE** – 1

@/songs/{ID} (user from session)

**GET – 5**

@/songs/{song-ID}

* //Retrieve the mp3 file for the song

@/songs/{song-ID}/info

* //Return information for {song-ID}

@/songs/liked (user from session)

* //Return all liked songs for current user

@/songs/{username}

* //Return all songs for {username}

**PUT – 1**

@/songs/{song-ID}/set-like-status?**value**=<1 / 0 / -1> (user from session)

* // Set the current user’s like status for song {ID} according to the request parameter
* @RequestParam int **value**
* **value**=1 -> like song
* **value**=0 -> set song to neutral
* **value**=-1 -> unlike song

**3.CommentController - 5**

**POST –** 1

@/comments

* song-ID
* comment body

**DELETE –** 1

@/comments/{comment-ID} (user from session)

**GET –** 2

@/songs/{song-ID}/comments

@/comments/{comment-ID}

**PUT – 1**

@/comments/{comment-ID }/set-like-status?**value**=<1 / 0 / -1> (user from session)

* // Set the current user’s like status for {comment-ID} according to the request parameter
* @RequestParam int **value**
* **value**=1 -> like comment
* **value**=0 -> set comment to neutral
* **value**=-1 -> unlike comment

**4.PlaylistController - 7**

**POST –** 1

@/playlists (body – name, user from session)

**DELETE –** 2

@/playlists/songs (user from session)

@/playlists/{playlist-ID} (user from session)

**GET –** 3

@/playlists/{playlist-ID }

@/users/{username}/playlists

* Return all playlists by user

**PUT - 1**

@/playlists/songs/ (user from session)

@/playlists (user from session)

**DAOs**

1. UserDAO
2. SongDAO

**POJOs**

1. User
2. Song
3. Comment
4. Playlist

**DTOs**

1. LoginDTO (username, password)
2. RegisterDTO (username, password, email, age)
3. UpdateDTO (newPassword, newEmail, newAge, newCity)
4. SongDTO (title, description, tags, binary song file)
5. FilterUserDTO (bySongs, byComments, byFollowers)