Music Discovery



Lyon Berends

Klas: DHI1V.Sa

Table of Contents

[1. Class diagram 3](#_Toc71106369)

[2. GET requests 4](#_Toc71106370)

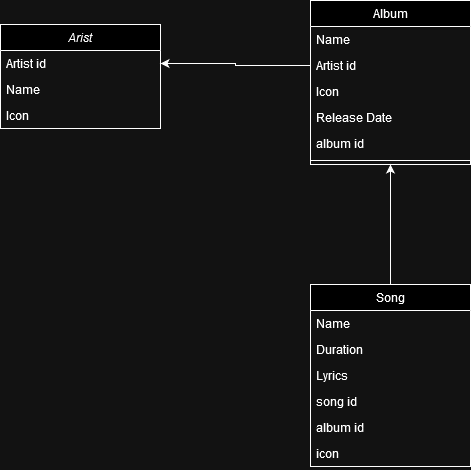
[3. POST requests 5](#_Toc71106371)

[4. PUT requests 6](#_Toc71106372)

[5. DELETE requests 7](#_Toc71106373)

Introduction  
This website will offer a nice music discovery of much which I recommend people listen to, this website will contain albums, songs and artists. You can add new albums, songs and artists using the live website and they will be presented on a webpage.

# Class diagram



Alle classes hebben een unieke ID.

De albums hebben een reference naar artists in de vorm van een artist id

De songs hebben een reference naar albums in de vorm van een album id

# GET requests

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/api/artists/{id}** | | | |
| Gets one artist, based on given id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *\*required* | | Id \* | path | Id of artist to find, returns all artists if empty |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Sends the artist class as JSON of 1 artist | |
| 404 | Sends JSON with 404 error code | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/api/albums/{id}** | | | |
| Gets one album, based on given id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *\*required* | | Id \* | path | Id of album to find |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Sends the album class as JSON of 1 album | |
| 404 | Sends JSON with 404 error code | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/api/songs/{id}** | | | |
| Gets one songs, based on given id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *\*required* | | Id \* | path | Id of song to find |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Sends the song class as JSON of 1 song | |
| 404 | Sends JSON with 404 error code | |

# POST requests

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POST** | **/api/artists** | | | |
| Adds an artist with the given name, albums and songs will be empty, id is automatically generated | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| * *required* | | Name \* | body | The artist is added. Could be send in JSON or Plain based on the content-type header |
|  | | Icon | Body | The icon of the artist |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 201 | Created successfully, will send a response in plain or JSON based on the content-header with a response code | |
| 500 | If something goes wrong with executing this request it will send a JSON or plain text response with the error code. | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POST** | **/api/albums** | | | |
| Adds an album with the given name, only JSON will be accepted | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| * *required* | | Name \* | body | The album name |
| * *required* | | Artist \* | body | The artist id which has to be linked to the album |
| * *required* | | Release date \* | body | The release date of the album |
|  | | Icon \* | body | The icon of the album |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 201 | Created successfully, will send a response in plain or JSON based on the content-header with a response code | |
| 500 | If something goes wrong with executing this request it will send a JSON or plain text response with the error code. | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POST** | **/api/songs** | | | |
| Adds a song with the given name, duration and lyrics | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| * *required* | | Name \* | body | Name of the songs |
| * *required* | | Duration | body | Duration of the song in seconds |
|  | | Lyrics | body | Lyrics of the song, not required, can also be added later. |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 201 | Created successfully, will send a response in plain or JSON based on the content-header with a response code | |
| 500 | If something goes wrong with executing this request it will send a JSON or plain text response with the error code. | |

# PUT requests

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PUT** | **/api/artists/{id}** | | | |
| Modifies an artist based on the given ID with a JSON body | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *\*required* | | Name \* | body | The new name of the artist |
| Icon \* | body | The new icon of the artist |
|  | | Id \* | Path | The artist id of the artist to modify |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Artist successfully modified, returns the new artist JSON in the body. | |
| 400 | Content type is not JSON | |
| 404 | Artist not found, cannot modify a non existing artist. | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PUT** | **/api/albums/{id}** | | | |
| Modifies an album based on the given ID with a JSON body | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *\*required* | | Name \* | body | The new name of the album |
| Release date\* | body | The new release date of the album |
| Icon \* | body | The new icon of the album |
| Artist id \* | body | The new artist id of the album |
|  | | Id \* | path | The album id of the album to modify |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Album successfully modified | |
| 400 | Content type is not JSON | |
| 404 | Artist not found, cannot modify a non existing artist. | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PUT** | **/api/songs/{id}** | | | |
| Modifies an song based on the given ID with a JSON body | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *\*required* | | Name \* | body | The new name of the song |
| Duration \* | body | The new duration of the song |
| Lyrics \* | body | The new lyrics of the song |
| Icon \* | body | The new icon of the song |
| Album id \* | body | The new album id of the song |
|  | | Id \* | Path | The song id of the song to modify |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Album successfully modified, returns the new album JSON in the body. | |
| 400 | Content type is not JSON | |
| 404 | Artist not found, cannot modify a non existing artist. | |

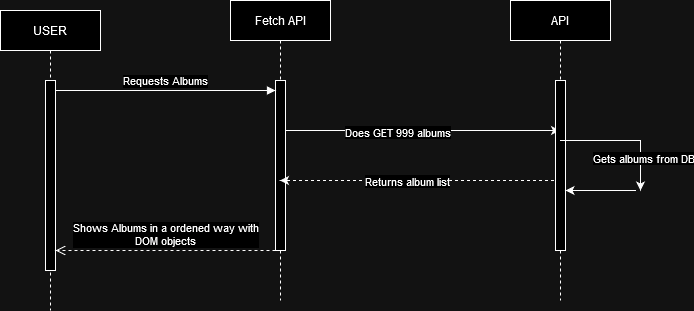
# DELETE requests

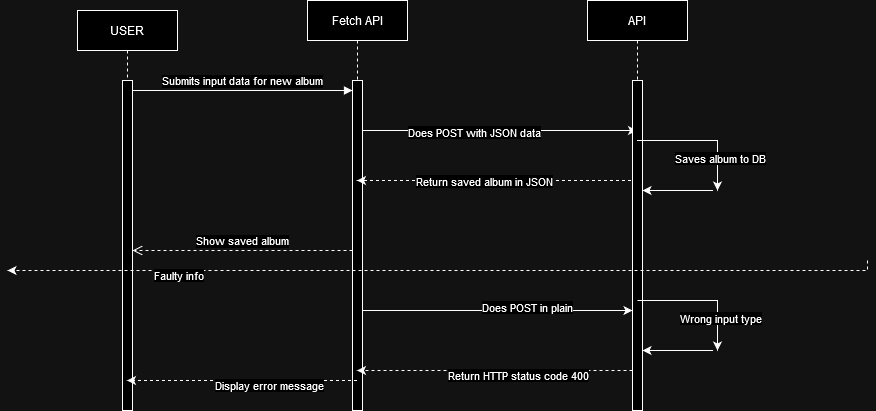
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE** | **/api/artists/{id}** | | | |
| Deletes one artist, based on given id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| * *required* | | Id \* | path | Id of artist to delete |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Artist successfully deleted | |
| 404 | Artist not found | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE** | **/api/albums/{id}** | | | |
| Deletes one album, based on given id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| * *required* | | Id \* | path | Id of album to delete |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Album successfully deleted | |
| 404 | Album not found | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE** | **/api/songs/{id}** | | | |
| Deletes one song, based on given id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| * *required* | | Id \* | path | Id of song to delete |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Song successfully deleted | |
| 404 | Song not found | |

# Sequence diagrams





Test report  
To make sure the website work in the way it was supposed to I tested every function on the website myself, I walked through all the processes which you are able to use on the website like editing, posting and deleting data. I made changes to the site based on the results of these tests, for example: the class diagrams have been changed based on the results of my tests, the case here was that an object couldn’t have an array of foreign keys.

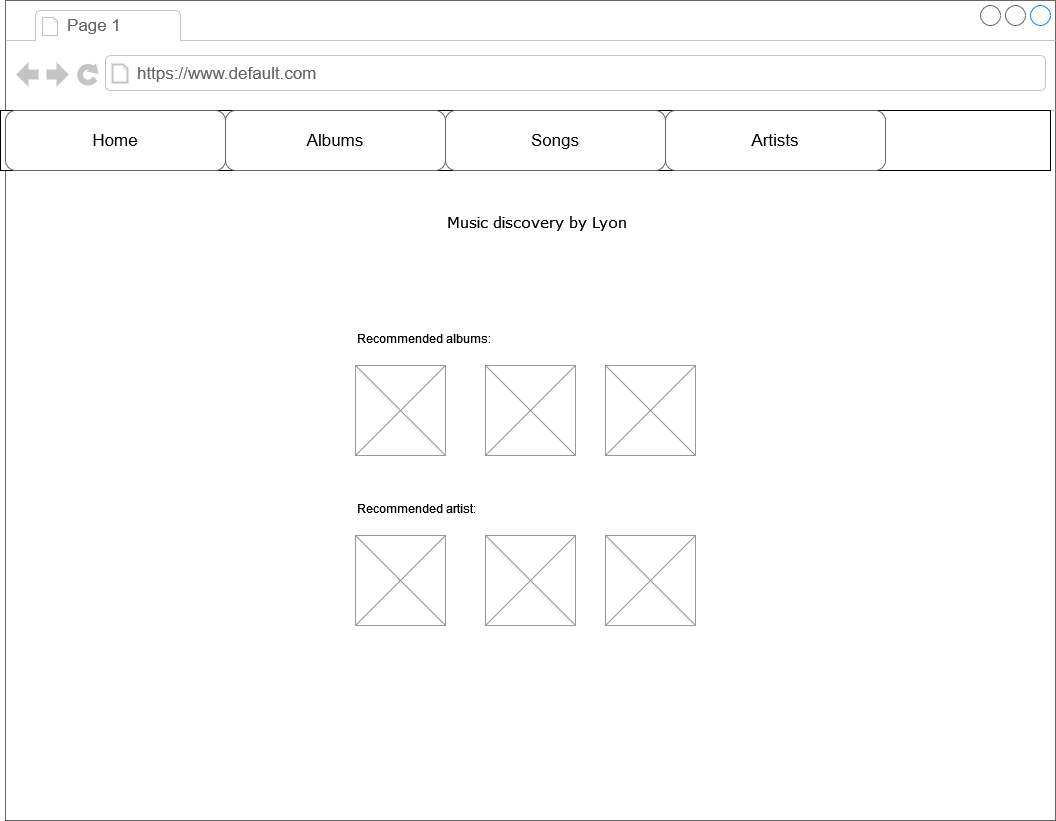
# Usecase diagram

Afbeelding met tekst, diagram, schermopname, lijn

Automatisch gegenereerde beschrijving

# Wireframes

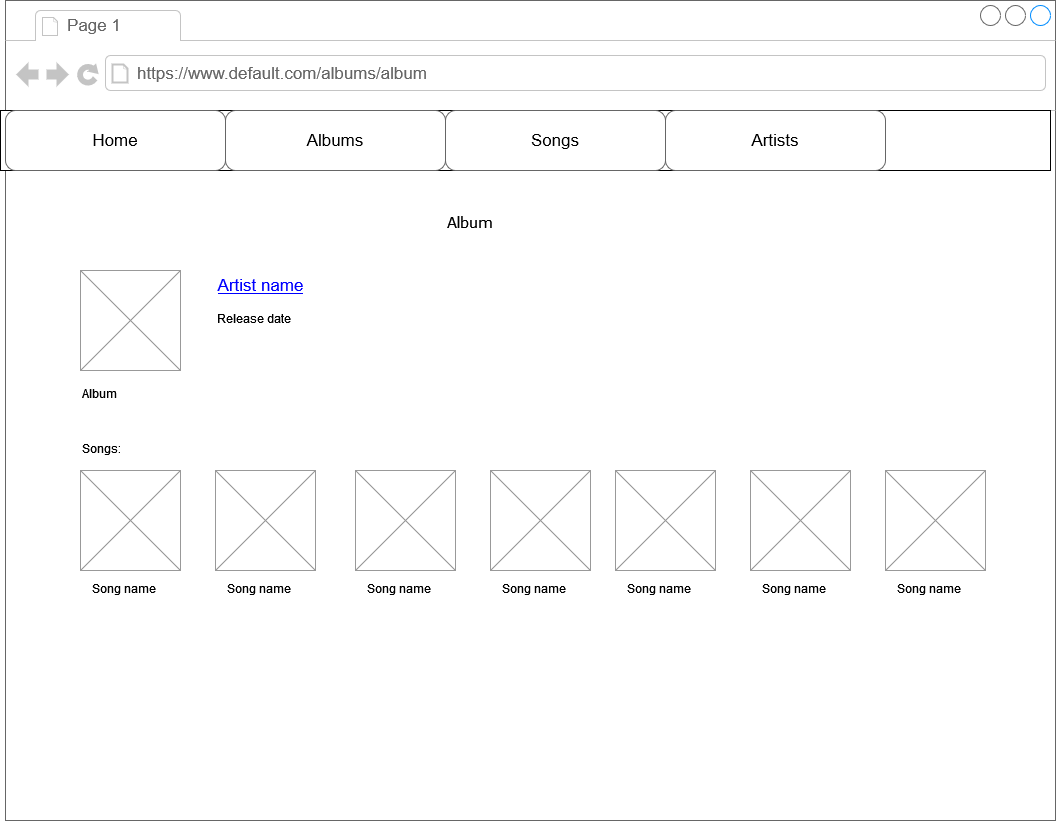
Index



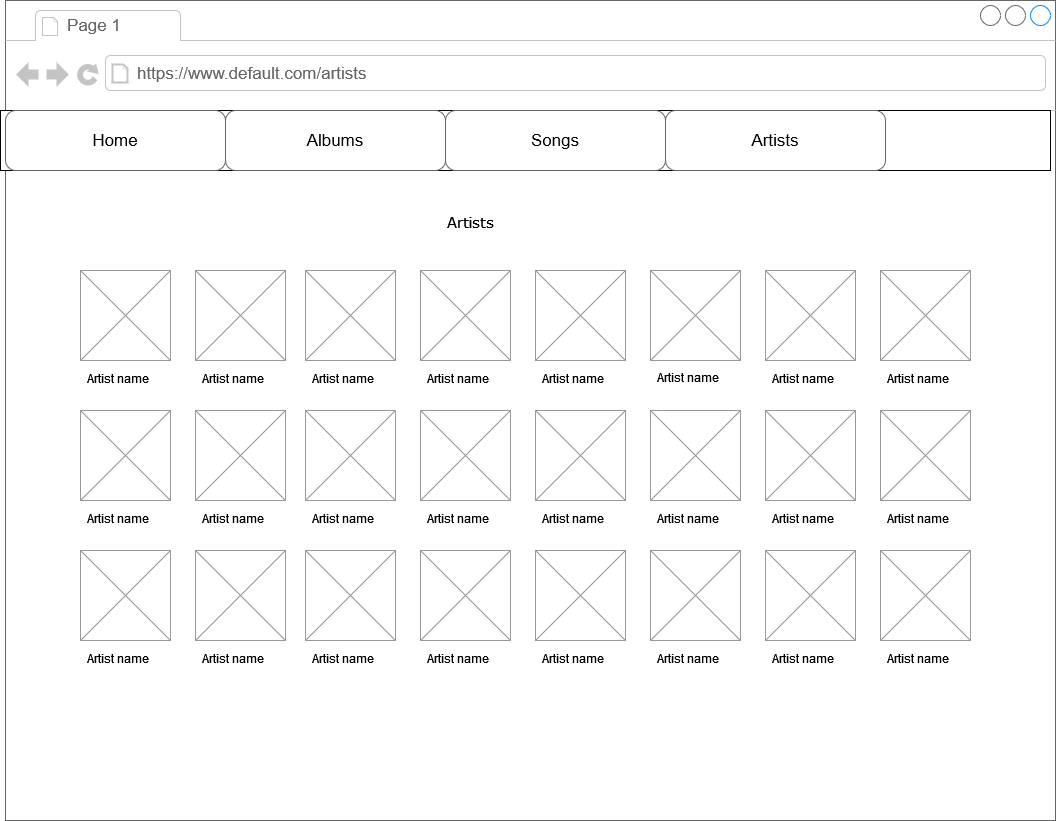
Albums



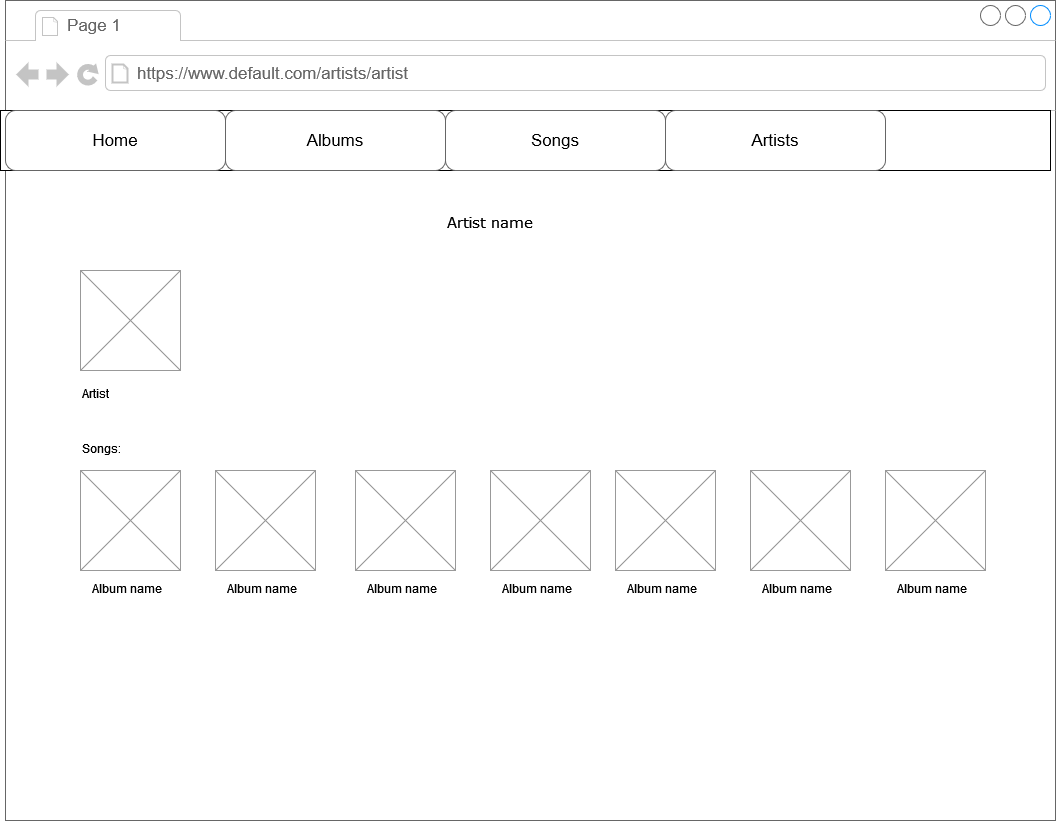
Album



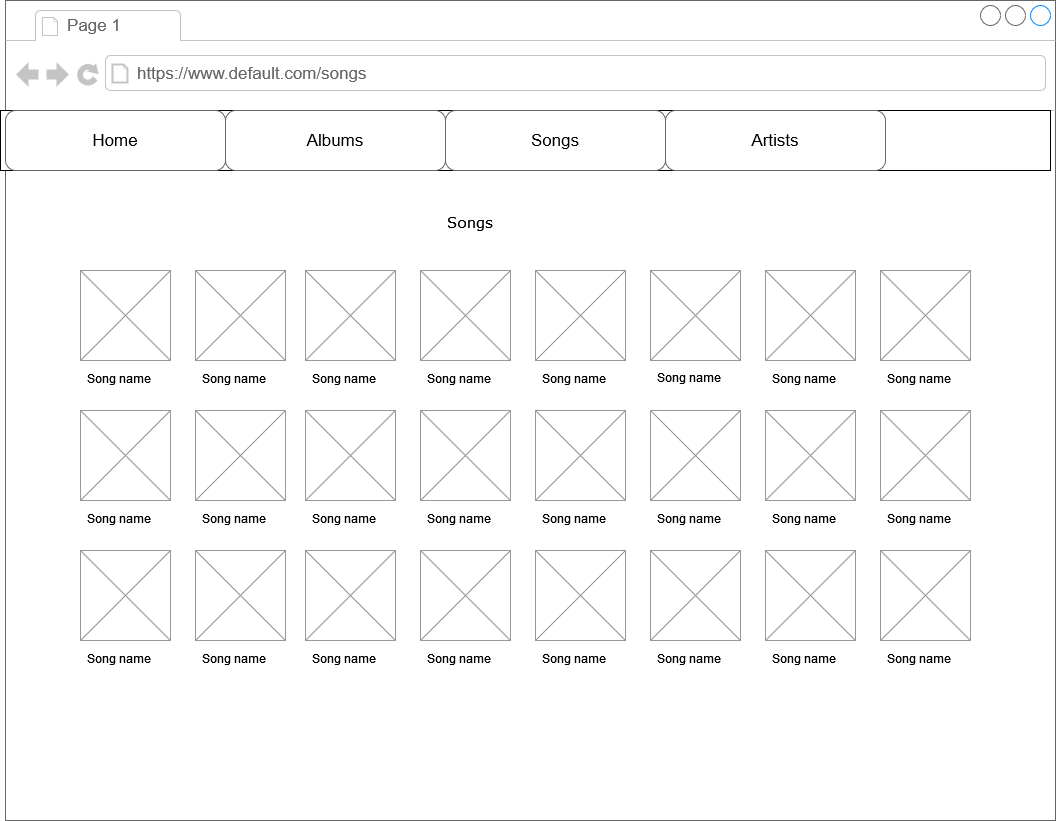
Artists



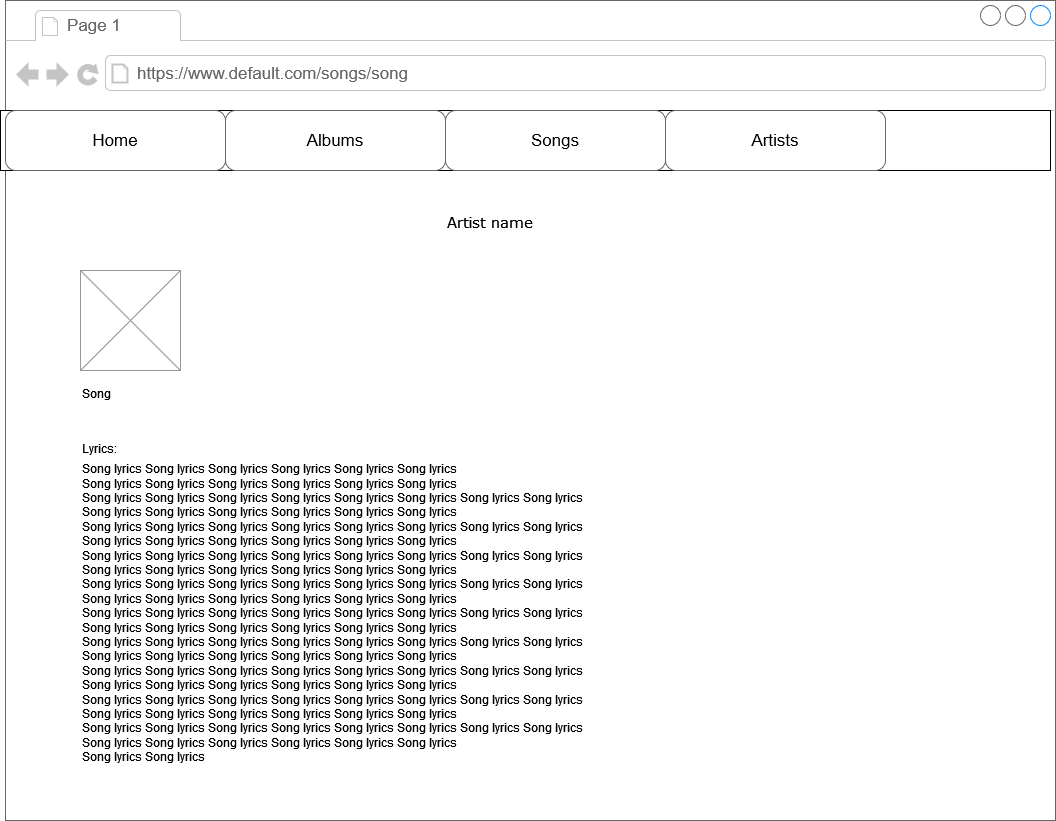
Artist



Songs



Song



## Explanation

The wireframes differ a bit from the end result, this is because when I made the wireframes I didn’t take in account that the ability to edit and delete data should also be a feature of the website, this has later been added but has not been updated in the wireframes.