

# Mobile App Development - iOS Exercise 1

> So spannend kann Technik sein.



# Abstract

**Build a simple iOS Application, which uses the “Last.fm” JSON API to request the “Top Albums” for a given “Artist”.**

## Modalities

Use a single view application template, with ARC & Storyboard.

Either use the standard `NSURLConnection` Class or the 3rd-party `ASIHTTPRequest` Library to make web requests.

Use a background thread to obtain and parse data from the “Last.fm” API.

Create two `UITableViewController`s, one for displaying (A) the data, the other for obtaining (B) the query information (user input, e.g. “Artist”).

- a. The display `ViewController` (A) needs to contain 3 `UILabel`s: “Album Name”, “Playcount” and “Artist”, as well as a `UIButton` (“Query Artist”) that will open the obtaining `Controller`.
- b. When the user presses the “Query Artist” Button, `ViewController B` is presented “modally”. This `ViewController` needs to contain one `UITextField` “Artist Name”. and a submit Button. When the submit Button is pressed, `ViewController B` provides the input data to `ViewController A`, which starts the web request, parses the json-response and displays the result (first “Album Name”, “playcount” of album and “Artist Name”).

# Additional Information

The “Last.fm” API endpoint for obtaining the “TopAlbum” for a given Artist is defined as:

[http://ws.audioscrobbler.com/2.0/?  
method=artist.gettopalbums&artist=ARTIST\\_NAME&api\\_key=API\\_KEY&fo  
rmat=json](http://ws.audioscrobbler.com/2.0/?method=artist.gettopalbums&artist=ARTIST_NAME&api_key=API_KEY&format=json)

The URL takes the following parameter (as seen in the example above):

```
method = artist.gettopalbums  
artist = Artist Name (example: “Joy+Division”)  
api_key = a65d06953de344d726c8f5a324f2aaad  
format = json
```

The API result is a json, containing the “Top Albums” for the given artist in an array (only display the first album):

```
{  
  "topalbums": {  
    "album": [  
      {  
        "name": "Album Name",  
        "playcount": 1234  
      },  
      { .. }  
    ]  
  }  
}
```

# Additional Information

Use a delegate pattern to pass information from ViewController B to ViewController A.

Use a UINavigationController to simplify the navigation stack.

Use either NSURLConnection or ASIHTTPRequest to make the web-request.

Use either -performSelectorInBackground or NSOperationQueue to process the data in the background.

Use either NSJSONSerialization or JSONKit to parse the json-data.

To replace all whitespaces with plus-signs in NSStrings, use the “stringByReplacingOccurenceOfString withString” function.

```
[@"artist name" stringByReplacingOccurrencesOfString:@" " withString:@"+"] ;
```

# Grading

Storyboard Setup: - Two viewControllers - Segues - View elements (buttons, labels, textFields) - Delegate pattern	3 PT
Asynchronous WebRequests	2 PT
Background-thread to parse the data	2 PT
JSON Parser	2 PT
Application Layout & Software Design	1 PT