Group 6 – Erwin Nieuwlaar, Celia Rodríguez and Yuliia Zavalniuk


# Project documentation

## Introduction

In the following page will be explained why each technology that was used in the project was the best option.

These are the following technologies that were used:

- Django
- Django Rest
- SQLite
- Libraries
- Bootstrap

## Django

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. It takes care of much of the hassle of Web development. It's free and open source. Django is considered to be one of the best options for back-end development using Python.

## Django rest

Django REST framework is a powerful and flexible toolkit for building Web APIs.

It is connected to the Django and has the best cooperation and interaction with it which allows working with different built-in or plugged-in features without caring about possible mismatches.

- The Web browsable API is a big usability win for your developers.
- Authentication policies including packages for OAuth1a and OAuth2.
- Serialization that supports both ORM and non-ORM data sources.
- Customizable all the way down - just use regular function-based views if more powerful features are not needed.

- Extensive documentation, and great community support.
- Used and trusted by internationally recognized companies including Mozilla, Red Hat, Heroku, and Eventbrite.

### SQLite

SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine. SQLite is the most used database engine in the world. The SQLite file format is stable, cross-platform, and backwards compatible. The Django framework use SQLite for storing data by default. There were no reasons to use another tool for this purpose.

### Libraries and packages

- Django:
  - django.shortcuts

    This module collects helper functions and classes that "span" multiple levels of MVC. In other words, these functions/classes introduce controlled coupling for convenience's sake.
  - django.db

    This module was used for working with database models
  - django.views

    This module was used for working with view function
  - django.http

    This module was used for working with Http requests
- rest_framework

  Django REST framework is a powerful and flexible toolkit for building Web APIs.
- rest_framework_csv

  Package which provides CSV renderer support for REST framework.
- Json

  Python built-in package, which can be used to work with JSON data.
- Statistics

Provides functions for calculating mathematical statistics of numeric data.

### Bootstrap

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Bootstrap 4 is easy to use and makes building a responsive front-end easy. The latest bootstrap version was used for providing clean representation of the data points of the data provided by Million Songs dataset.

### To start working

All of the API sample requests and functionality explanations can be found in the API documentation (Milestone 1).

The web service itself has an intuitive, user-friendly interface with the navigation bar, search bar and multiple links to other pages. "Music" button opens the list of all artists sorted in alphabetical order. Information about each artist can be opened after clicking on the link. On every informational page about the artist, there is information about artists and links to the list of artist's songs and google searches.

Also every artist detail page contains links to the pages with statistics (mean, median, standard deviation). Every page with statistics contains links to the other types of statistics calculations and an optional list of year where particular artist recorded the songs. Choosing one of the year option will open the page with statistics for particular year.

List of songs contains links to the informational pages of the specific songs, additional information about artists and links to the other songs of the artist and google search of the artist.

After choosing Sort by -> Artist by genre option the list of all genres opens, where it's possible to click on the genre and open the list of the artists with perform in this particular genre.

After choosing Sort by -> Artists by popularity option the list of all artists opens, with the possibility to subset the result.

After choosing Sort by -> Songs by popularity option the list of all songs sorted by popularity opens.

After choosing Sort by -> Songs by years option the list of all years when the songs were recorded opens. After choosing one of the years the list of songs in that specific year opens.

The search box allows the user to search for data. The results will contain all of the songs which contain the keyword in its artist_name, artist_terms or song_year fields.

All of the pages can be rendered into .json or .csv format. Also, there is a possibility to download the whole dataset.

The dataset should be imported into the system in .csv format with required fields.