

# Background

Banks are responsible for stopping money laundering. Hence it is necessary to avoid money transfers to terrorists and criminals. A list of such people is published by the EU.

## Task

Your task is to implement a simple web service that compares a given name against a list of sanctioned people to detect suspicious transfers.

Your API should have the following functionality:

1. Web service that takes a name as an input and returns whether a name match exists. The returned value can be a boolean, but can also contain extra data if needed.
  - a. The web service should be able to handle noise words (example: the, to, an, mrs, mr, and) that have to be ignored.
  - b. The web service should be able to detect names with simple spelling errors and also return as few "false positive" matches as possible.
2. Web services to add, update and remove sanctioned names.

## Technical requirements

- Implement using Java and Spring framework.
- Sanctioned names should be stored in a database. (In-memory database can be used.)
- Open source libraries can be used.
- Source code should be stored in a git based version control system.

## Example

There is a sanctioned person with the following name: "Osama Bin Laden". Out of the box, your solution should flag at least the following user entries as possibly suspicious:

- "Osama Laden"
- "Osama Bin Laden"
- "Bin Laden, Osama"
- "Laden Osama Bin"
- "to the osama bin laden"
- "osama and bin laden"

If you find the list above too simple, then you can try implementing an algorithm that could match the following user entries as well:

- "Ben Osama Ladn"
- "Ladn the Asoma"

In reality, name matching is not a trivial task. Matches could also happen based on:

- Substring matches - sl: "Robert", user: "Bert"
- Ignoring spelling errors - sl: "Madis", user: "Madus"
- Abbreviations - sl: "Joe Luis Webb", user: "Joe L. Webb"
- Noise words - sl: "Mr. John Smith" user: "John Smith"