Programming assignment Python

Assignment

Create a service in Python that receives information about 2 people and calculates the probability that those 2 people are the same physical person. Person is represented by the following attributes:

- First name
- Last name
- Date of birth
- Identification number

The matching logic should work in the following way:

- If the BSN number matches then 100%
- Otherwise:
    - If the last name is the same +40%
    - If the first name is the same +20%
    - If the first name is similar +15% (see examples)
    - If the date of birth matches + 40%
    - If the dates of birth are known and not the same, there is no match

Similar first names examples:

- Andrew and A. (initials)
- Andrew and Andew (typo)
- Andrew and Andy (diminutive)

Matching Examples:

Person 1	Person 2	Matching result
Andrew Craw 20-02-1985 BSN unknown	Andrew Craw BSN unknown	60%
Andrew Craw 20-02-1985 BSN unknown	Petty Smith 20-02-1985 BSN unknown	40%
Andrew Craw 20-02-1985 BSN unknown	A. Craw 20-02-1985 BSN unknown	95%
Andrew Craw 20-02-1985 BSN 931212312	Petty Smith 20-02-1985 BSN 931212312	100%

Further requirements

- Use the authentication mechanism of your choice (like basic authentication, api key, or any public identity provider like Facebook or Google, credentials can be hardcoded)
- The API should follow RESTful principles and return meaningful HTTP status codes for all cases

Acceptance criteria

- Functional completeness and correctness

FRISS

Orteliuslaan 15
3528 BA Utrecht
The Netherlands
+31 30 767 0352

Kvk: 30217641
BTW: NL 817.067.085.B.01
IBAN: NL85ABNA0553612174

- Use of the appropriate tools, web framework idioms and standard methodologies
- Proper REST API design
- Unit tests

Bonus points for

- Documentation for the API
- Developing the web UI interface as a full SPA with your client side framework of choice
- Server side caching
- Integration tests
- Dockerfile
- Logging for HTTP incoming requests

What to deliver

1. An archive containing the code or a link to a git repository. Note that required modules should restore without problems
2. Example of an API call
3. A readme file describing how to run the application

Final Note

This is a chance for you to shine and showcase your technical skills, write the best code you could, pay attention to details which you find the most important and make sure you use the latest and greatest of the technologies required. This is a timeboxed exercise, we understand you cannot finish everything on time, so please focus on what you believe is important. Good luck.😊