**Option: Using the Spring Boot Service from Flask**

If you want to **reuse the existing Spring Boot service** (without rewriting it in Flask), you can call the Spring Boot service from Flask using HTTP requests.

To do this, you could use Python’s requests library to call the Spring Boot API:

1. Make sure your Spring Boot email service is accessible through an HTTP endpoint (e.g., via a REST API).
2. In your Flask app, use requests to send a POST request to your Spring Boot service to trigger the email:

python

CopierModifier

import requests

def send\_confirmation\_email\_via\_spring\_boot(to\_email, name, token):

url = "http://localhost:8080/send-email" # Replace with the correct URL of your Spring Boot endpoint

payload = {

"toEmail": to\_email,

"name": name,

"token": token

}

try:

response = requests.post(url, json=payload)

response.raise\_for\_status()

except requests.exceptions.RequestException as e:

print(f"Error sending email via Spring Boot service: {e}")

This way, you won't need to re-implement the email functionality in Flask, and you can still use

Haut du formulaire

Bas du formulaire