VOTING AGE CALCULATOR - LQA TOOL DEMO QUESTION

Design **Voting Age Calculator** Application with following folder structure -

A. PROJECT STRUCTURE

1. Create a folder with name '*agecalculator*'.
2. '*agecalculator*' folder should have 2 sub folders
    1. First sub-folder is '*core*'.
    2. Second sub-folder is '*validations*'
    3. Followed by a *service.py* file.

B. CONTENTS OF PACKAGES

Contents of core sub-folder:

- *__init__.py*
- *CalculateEligibility.py*

Contents of validations sub-folder:

- *__init__.py*
- *Validators.py*

C. QUESTION DESCRIPTION

1. **core/CalculateEligiblity.py**
    1. Create a function compute_eligibility with 2 arguments (name and age)
    2. This function should call name_validity_checker(name) from validations/ Validators.py
        1. If it returns False, return "Username is invalid".
        2. If it returns True, call age_validity_checker(age) from validations/ Validators.py
            1. If it return False, return "Age is invalid"
            2. Else, check whether the user is eligible to vote! (If user age > 18, return "Yes", else return "No".
2. **validations/Validators.py**
    1. Create name_validity_checker(name) function, it will return True if length of name is >=5, otherwise False.
    2. Create age_validity_checker(age) function, it will return True if age is positive, else False.

3. service.py

1. Create a function compute_result() and get inputs for name and age, then call compute_eligibility() function and test your result.

Important Notes:

1. Follow proper naming conventions and standards as per question.
2. Do not get any inputs from user. Make sure that, your application is free from infinite loops.
3. Your code must execute within 5 seconds.
4. Make sure to return proper messages / return types as per the question.

Git Commit Guidelines:

1. Inside your *talentpy* repo, submit your project.
2. Ensure https://github.com/repo_name/talentpy lists your project and must be PUBLIC for evaluation.

Sample Input/Output:

1. compute_result() => name = “Hari”, age=14 : Expected Output: Username is invalid.
2. compute_result() => name = “Hariprasath”, age=-3 : Expected Output: Age is invalid.
3. compute_result() => name = “Hariprasath”, age=19 : Expected Output: Yes
4. compute_result() => name = “Hariprasath”, age=12 : Expected Output: No