Code Challenge

Write a .NET RESTFul webservice for below two application scenarios

SCENARIO 1

ABC company stores their employee details in below CSV files.

File Name	Content	Values Order
employee.csv	Comma separated employee details	Employee Id, Full Name, Gender, Date of Birth, Joined Date, Salary (USD), Branch
currency.csv	Comma separated currency details. Rates are given from USD to the relevant currency.	Country, Type, Rate

- Above csv files are provided to you.
- You can't change the file names and the file contents.
- Write a rest API to return salary details. Get an employee Id as a user input.

Salary detail should contain

- Employee Id
- Full Name
- Branch
- Salary (In local currency - E.g. LKR: 100,000.00)
- PATE Tax Amount (In local currency - E.g. LKR: 1,000.00)
- Net Pay Amount (take home after all deductions in local currency - E.g. LKR: 99,000.00)

SCENARIO 2

● Develop a search API to support below criteria

- Search Terms (Country, First Name, Last Name, Branch)

- Minimum one term should be presented

- LIKE search should be supported for First Name and Last Name if (* entered at the end of the search term).

Other search terms don't need to support LIKE Search.

- Case insensitive all four search terms.

- If multiple terms are entered consider it as a AND

- e.g country = Sri Lanka AND First Name = Clement AND Last Name = Fernando AND branch = Colombo

Please note that this API endpoint should support returning 1000 records and any best practices you are following to cater to the load will be considered as your technical excellence.

- For testing purposes, you can edit the csv file and include test data without modifying the existing ones

PAYE Tax Rate:

Country	Range (After converting to local currency)	PAYE Tax Rate
Sri Lanka	Below 100,000	0%
	100,000-249,999.99	5%
	Above 250,000	10%
India	Below 100,000	0%
	100,000-299,999.99	4%
	Above 300,000	7%
Pakistan	Below 500,000	0.5%
	Above 500,000	4%
Bangladesh	No Tax	

Your rest API should support below http codes when responding

- 200 for success case with success message and above salary detail as a data
- 400 for invalid Employee no with error message and no data
- 500 for internal server errors with error message and no data

SCENARIO 3

Please write a frontend web application using (Angular 10 - preferred / React JS, Bootstrap, HTML, CSS3) to support below scenario.

Build a dummy Login page with a hard-coded username and password to login to the application. Once logged in, the employee details page should consist of a search criteria to facilitate the Scenario 2 API requirement which can be used to search employees render it. Visualization of the following could be creativity of your own thoughts. Note your solution should be scalable to search more than 1000 records.

Hints – Lazy Loading or Pagination

Sorting

OAuth 2.0

EVALUATION CRITERIA (IMPORTANT)

We will look at your project and assess it for:

1. Extensibility - separation of concerns.
2. Simplicity - aim for the simplest solution that gets the job done whilst remaining readable, extensible and testable.
3. Test Coverage - breaking changes should break your unit tests. Mock CSV contents. No need to write unit tests for the file reading.
4. Performance - should gracefully handle a significant increase in amount of data provided (e.g. 10000+ employees).
5. Robustness - should handle and report errors. If you have any questions about this criterion, please ask.
6. Use of appropriate data structures
7. Proper error handling

SPECIFICATIONS

.NET Framework , C# , Angular or ReactJS

Include a README with (accurate) usage instructions.

SUBMISSION

Github is the preferred option (a public repo is fine) but we will also accept a .zip file if necessary. Email your solution to us.