Coding Exercise – Projects Tracker (2 hours)

Objective

Develop a search mechanism for employees allocated to projects that will allow the HR to search, track and view all employees and which project they are presently working on. The exercise is supposed to have SOAP or REST endpoints, so that it can be integrated with any external system. We expect you to use the latest .Net Framework or .Net Core to develop this functionality.

All API endpoints should adhere to basic .Net coding standards and should be designed keeping in mind reusability and extensibility. Ideally, we would like this to be API endpoints, but you could also implement the same functionality as a console application adhering to the same principles, if required.

Step 1: Create Database Structure

Create a table structure that could store the following data. You can make assumptions regarding size of each of the fields. The data provided below is only indicative and the actual data may run into millions of records.

Name	Company	Project	Role	RecordID
Ankur	IBM	Payroll	Software Engineer	1
Akash	IBM	Chat Bot	Software Engineer	2
Priya	HP	VR Gaming	Project Manager	3
Asha	Microsoft	Payroll	Solution Architect	4
Nandini	HP	Payroll	Software Engineer	5
Piyush	Microsoft	Payroll	Delivery Manager	6
Ankur	HP	Chat Bot	Lead Engineer	7
Akash	HP	VR Gaming	Software Engineer	8
Priya	IBM	Payroll	Solution Architect	9
Asha	HP	Chat Bot	Project Manager	10
Nandini	IBM	VR Gaming	Lead Engineer	11
Piyush	Microsoft	Chat Bot	Delivery Manager	12

Step 2: Search API

Create a new search end point, which supports AND, OR and NOT operators. The results are in JSON format. For example:

query=Piyush AND Microsoft should return only all the details from Record ID 12.
query=Ankur AND NOT HP should return all details from Record ID 1

The AND operator has a higher precedence than OR, but you can use parentheses to override this behaviour.
query=Software Engineer AND (Payroll or HP or Nandini) should return records 1, 5 and 8.

Use double quotes for exact searches. For example:
query="Chat Bot" and Ankur

Please note that all searches are always case insensitive.

Extra Credit: Along with search keywords, support the use of the following syntax as well.

AND: ,
OR: |
NOT: -

For example,
query=Piyush, Microsoft
query=Ankur, -HP
query=Software Engineer, (Payroll | HP | Nandini)

Step 3: Pagination

Allow the search to not return all the information but implement pagination. On passing PageSize and PageIndex into the search input, only applicable results will be returned. For example,

query=Payroll&PageSize=2&PageIndex=2

This will return records 5 and 6 only. Adding pagination will also add the following additional return values,

total – total number of matching results

pageSIze – the page size passed by the calling party

currentPage – the page index passed by the calling party

pages – total number of result pages available based on the page size and total matching results

General Guidelines

- Use SOAP/Rest endpoints, if possible
- Use API URL conventions where applicable
- Full text search is implemented keeping in mind performance against millions of records
- The search implementation is easily extensible to additional columns at later stage
- Search lingo is easily extensible at a later stage
- Code is structured such that its easy for others to understand

Note

Relax, have fun and put your personality into your design and code. Beg, borrow and steal from the internet because in reality this is how we all write software. Do understand what you are borrowing from the internet. And remember, non-completion of the exercise is OK, because we'll be principally examining how you approached the problem, and not necessarily the end result!

Good luck!