Exercise

BOOK SEARCH

EXERCISE

The exercise should be implemented in Microsoft .NET and consist of two parts. You should develop the web application in Visual Studio. Feel free to use any information available on the internet when developing your solution. Attached with the exercise is an XML-file (books.xml) that should be used as data source. The books.xml file must be included and used in your solution. The source files must be included along with a description that explains how to run your solution.

PART ONE – WEBSERVICE

Implement a webservice API that can be used to search for books. Use the books.xml file as your data source. The file contains a set of books with the following properties:

- **author** (type: string): The author of the book
- **title** (type: string): The title of the book
- **genre** (type: string): The books genre
- **price** (type: decimal): The price of the book in USD
- **publish_date** (type: datetime): The publishing date of the book
- **description** (type: string): A short description of the book

Each book is defined in a book-tag with an **id** (string). All books are encapsulated in a **catalog**-tag which is the root node of the XML-file.

The API should, at least, contain a method to search for books by **title**. The method should support partial searches.
Example: A search for "Ruby" must include both "Deploying with JRuby" and "Build Awesome Command-Line Applications with Ruby" in the result.

PART TWO – WEBSITE

Implement a frontend website using the API from part one.

EVALUATION

The solution is evaluated based on code structure and code design. This is a way for us to measure your skill of .NET and practical use of the platform.

The following criterias should be kept in mind:

- **Transparency**: Is it possible to understand how the web application works simply by looking at the code?
- **Reasonable**: Are the techniques and design patterns chosen deliberately
- **Reusable**: Is the code developed with reusability in mind? Can parts of the code be reused?
- **Examplary**: Can the code be used as an example for similar problems?

Bonus is given for an extended search API and a modern website.

Feel free to impress us! Good luck!