Mitchell Coding Challenge

Do you want to work at Mitchell? Great! We want you here too. But first, we want to see your skills in action. The following is a programming challenge that will help us figure out your strengths and how we can best utilize your talents. If you would like to be considered for an intern position, please finish and return the coding challenge below.

We hope that you take this challenge seriously and work on it without the assistance of other developers. That said, developers don't work without resources, so feel free to leverage the internet or books if necessary.

Above all, this programming problem is less about IF you solve the problem and more about HOW. The elegance and extensibility of your solution is important. Let this be your chance to demonstrate your understanding of good software principles.

So enough talk – let's get to the fun stuff.

Coding Challenge

Required Components

Implement a solution that performs CRUD operations (Create, Read, Update, and Delete) for a Vehicle (automobile) entity.

A Vehicle is a simple object defined as follows:

```
public class Vehicle
{
    public int Id { get; set; }
    public int Year { get; set; }
    public string Make { get; set; }
    public string Model { get; set; }
}
```

You must implement the following methods:

GET vehicles
GET vehicles/{id}
CREATE vehicles
UPDATE vehicles
DELETE vehicles/{id}

Additionally any solution must employ the following:

1) Usage of either C# or Java.
2) Some form of automated testing.
3) Some form of in-memory persistence of created vehicle objects.

Optional

Got all the required stuff nailed down and want to enhance your implementation a little further? Here are some additional features you can add to make the implementation more interesting.

1) Add validation to your service.
    - Vehicles must have a non-null / non-empty make and model specified, and the year must be between 1950 and 2050.
2) Add filtering to your service.
    - The GET vehicles route should support filtering vehicles based on one or more vehicle properties. (EX: retrieving all vehicles where the 'Make' is 'Toyota')
3) Write an example client for your service.

Evaluation

Your implementation will be evaluated on the follow criteria:

1) Whether or not all methods are implemented.
2) The maintainability / readability of the implementation code.
3) The flexibility of the implementation code. How well does the code base adapt to changing requirements?
4) The testability of the implementation code.
5) The reusability of the implementation code. Can facets of the implementation code be used for other services or applications?

Submission Instructions:

Please submit your solution by emailing both sarah.murphy@mitchell.com and nitin.shitole@mitchell.com a DropBox or GoogleDrive link by 9:00am PST on 10/31/2016.

If you have any questions about the coding challenge feel free to email Nitin at nitin.shitole@mitchell.com. If you have any questions about the intern program or interview process, please feel free to email me at sarah.murphy@mitchell.com, or call me at 858-368-7473.

Good luck!