Exercise for Candidates

In this exercise we will ask you to implement a simple Address Book page. The exercise involves two major sections: One for the front-end, and one for the back-end implementation.

Front-End

The Mockup folder included in the package along with this document contains a raw HTML implementation of a very simple Address Book page. Within that implementation you will find a grid like structure that contains contacts. Even though that is the most important section, the entire web page is important.

For the front-end portion of the exercise, you are not expected to work in .NET at all. Instead you will work directly on the HTML interface that we have provided. You are strongly encouraged and expected to explore all HTML and CSS files and make them your own. Make any changes that you would normally do, according to the best practices that you follow, to improve the readability of the files, and to make it work with multiple browsers.

The files with your changes will be compared with the originals. None of your changes will go unnoticed.

Instructions

The list below contains a set of tasks you have to complete for the front-end portion of this exercise. This list contains: Requirements, bugs and a new request.

- REQUIREMENT: The information in the Email column must be visible in its entirety, but you can't reduce the width size of the Name and Phone columns; they already have their required fixed size.
- REQUIREMENT: All the buttons you implement on this exercise must have the same look and feel as shown below. Use fully colored buttons for the normal and inactive states, and gray for the hover state.

- BUG: The look and feel of these buttons must work on the most common browsers (IE, FF, Safari, and Chrome). There is a known issue for these buttons in Chrome. Apply a fix to make it work. (**Nice to have**)
- BUG: There's an error in the data grid when the user moves the cursor over the rows. The image below is the expected behavior.

- BUG: There is a problem with the buttons and grid effect when we scroll down the page, sometimes these controls are disabled for reasons unknown.

Section Complete

Once you have completed the tasks listed for the back-end portion of this document, you may use your completed UI to continue with the back-end portion below.

Back-End

In this section you will implement the CRUD operations of the Address Book using an ASP.NET MVC application. Use the front-end that you have already developed to maintain a list of contacts with their names, email addresses, and phones.

Besides following best-practices, design patterns, and standard naming conventions, there are some additional guidelines to follow:

1. Create the application in C#, and as mentioned earlier, use ASP.NET MVC.
2. The data will need to be persisted in SQL Server 2008 or above.
3. You are expected to use one of the following ORMs: Entity Framework, or NHibernate.
4. Use multiple projects to make the architecture as scalable and maintainable as you think is appropriate. At a minimum we expect a variation of the following projects:
    a. Web application project.
    b. Data project for database entities.
    c. Repository project.
    d. Interfaces for Repository project (**look for Repository Pattern**). For the purposes of this document, we refer to this project as IRepository, but you can name it however you want.
5. The Web Application should not depend on the Repository project directly; instead it will use the IRepository project, using IoC with the Framework of your choice. StructureMap is ideal (**easy to set up**).

6. As a new requirement, we need to register the vehicles that belong to each one of the persons in our contacts list. Each person might have zero or more cars; for each one, we will need a manufacturer, model, year, and license plate. (**Nice to have**)

You may include any documentation you want, if you think it's necessary to help us test your application. You are free to refactor the architecture of the application as you think is best, and use any other components, as long as their licenses allow it.

The **"Nice to have"** legends are pluses to the exercise but not required at all though.

Exercise Complete
Done and not tired yet? For extra credit, add an option to upload a picture for each car.