Please provide an ASP.NET MVC4 multi-tiered solution that is able to handle the following requirements (expected delivery time: 2 days):

|  |  |
| --- | --- |
| **1)** | illustrate the use of a masterpage. |
| **2)** | implement a web page that shows the employees of a company and their department in a grid. The user should be able to filter the data based on a person's name and/or a department selectable from a dropdown. Filling the data for the grid and for the drop-down should be done via AJAX calls. |
| **3)** | the web page from (2) should feature the possibility of adding a new employee into the database. This means that the user should see a special section on the page with fields in which they can fill in details about the employee (e.g. first and last names, phone number, email, date of birth, department). After saving the data, the grid on the page should be refreshed (via AJAX). |
| **4)** | the solution should offer support for tracing employee movement accross departments (i.e. *if necessary*, we should be able to easily implement a report showing for each employee the history of departments where they worked and the corresponding periods, so this should be taken into consideration when designing the solution). |
| **5)** | as domain object, the Department should feature at least: Name, ParentDepartment. |
| **6)** | as domain object, the Employee should feature at least: FirstName, LastName, Email, PhoneNumber, DateOfBirth, Department. |
|  | The following requirement is *not mandatory*, but will be considered a plus: |
| **7)** | the user should have the possibility of adding dynamically new attributes when registering a new employee. These attributes should later be queryable in case a report is required (i.e. all new attributes should be displayed as report filter fields, with appropiate value types - age should be a positive INT, salary should be a DECIMAL):     **\*** from the UI perspective, the user should be able to click a button ("Add...") for adding a new attribute beside the fixed ones already implemented.      **\*** a dropdown placed inline with the button should contain attributes described previously by other users. The first entry in the dropdown will always be "-new attribute-", followed by attributes saved previously.      **\*** after clicking the "Add..." button, the UI is updated with the necessary fields. In case the "-add new attribute-" option was selected, then the user will be shown the necessary fields for describing and adding a new attribute. In case some other option in the dropdown was selected, the user will be shown the corresponding field for entering the attribute value ensuring only the correct type of data is allowed. |
| **HINTS** | **-** use EntityFramework 6.x (code first) and MS SqlServer for persisting data. **-** in the context of a multi-tiered solution, we expect that there will be at least 2 layers containing data-objects (one of them being the EF data layer in which our entities reside). In order to convert (adapt) one such data object in a layer to its corresponding data object equivalent in another layer, please use *AutoMapper v3.3.1*. **-** the use of KnockoutJS is considered a plus. |