ASP.NET MVC Design – Übungen

Du kannst auf dem Stand der Basic-Übungen aufbauen oder das Source-Paket von <https://github.com/garaio/aspnetmvc-intro/zipball/master> herunterladen.

# Step 1: Erstelle eine ASP.NET MVC Applikation

* Erstelle die Applikation mit dem VS New Project Wizard (das Unit-Test-Projekt ist vorerst nicht nötig).
* Starte sie und stelle sicher, dass sie funktioniert.

# Step 2: Kontaktliste

* Erstelle eine Model-Klasse „Kontakt“ mit den üblichen Feldern (Vorname, Name, Adresse, PLZ, Ort)
* Erstelle einen Mechanismus, um Kontakte zu persistieren (in-memory, statische List<T> oder so)
* Erstelle beim Applikationsstart 2-3 statische Kontakt und speichere sie in der Liste
* Erstelle einen Controller „Kontakte“, der in seiner Index-Methode die aktuelle Kontaktliste aufbereitet und anzeigt.
* Erstelle eine View für die Kontaktliste. Zeige in der Liste nicht alle Attribute an.

# Step 3: Kontakt-Details anzeigen

* Füge jedem Eintrag der Liste einen „Details…“-Link hinzu
* Erweitere den Controller „Kontakte“ um die Methode „Details“, welche die Kontaktdetails aufbereitet und anzeigt.
* Erstelle eine View für die Kontaktdetails.
* Verlinke den Details-Link mit der neu erstellten Detail-Seite.

# Step 4: Kontakt-Details ajaxifizieren

* Verwende jQuery und fancybox (<http://fancybox.net>) (oder ein anderes Lightbox-Plugin Deiner Wahl), um die Details nicht auf einer neuen Seite, sondern als Lightbox darzustellen.
* Passe dazu die Details-View so an, dass die MasterPage nicht verwendet wird, sondern nur das erforderliche HTML-Fragment zurückgegeben wird.

# Step 5: Kontakte erfassen

* Erweitere den Controller „Kontakte“ um die Möglichkeit, einen neuen Kontakt zu erfassen.
* Validiere die Eingabedaten und zeige allfällige Fehler jeweils direkt beim entsprechenden Feld. Wenn alles richtig eingegeben wurde, so speichere den Kontakt.
* Verwende für die Anzeige der Fehler Html.ValidationMessageFor & Co.
* Verwende das Post-Redirect-Get-Pattern, um anschliessend wieder die Kontaktliste anzuzeigen.

# Step 6: Kontakte löschen

* Erweitere den Controller „Kontakte“ um die Möglichkeit, einen Kontakt zu löschen.
* Das Löschen soll über einen Link/Button auf der Detailseite erfolgen. Die Aktion muss bestätigt werden.
* Verwende das Post-Redirect-Get-Pattern, um anschliessend wieder die Kontaktliste anzuzeigen.

# Step 7: PLZ-zu-Ort-Eingabehilfe

* Erstelle einen Controller „PlzHelp“, der anhand einer PLZ mögliche Orte als JSON-Array zurückgibt.
* Verwende als Datengrundlage dafür <https://match.post.ch/downloadCenter?product=2> (PLZ Light)
* Ergänze die Eingabemaske mit dem erforderlichen jQuery/Javascript-Code, um beim Verlassen des PLZ-Felds einen möglichen Ort per AJAX zu ermitteln und in das Orts-Feld zu schreiben.