



Projekt: wt4u

Code Reviews

Andreoli Dario (dandreol@hsr.ch)
Schiepek Richard (rschiepe@hsr.ch)
Zahner Tobias (tzahner@hsr.ch)

Änderungsgeschichte

Datum	Version	Änderung	Autor
24.04.2014	1.0	Inititaldokument	rschiepe
25.04.2014	1.1	Alle Probleme, Verbesserungen aufgeführt	rschiepe
15.05.2014	1.2	CodeReview C3 Probleme/Verbesserungen	rschiepe
19.05.2014	1.3	Erweiterungen CodeReview C3	dandreol

Inhalt

Änderungsgeschichte	2
Inhalt.....	3
1. CodeReview C2	4
1.1 Strategie.....	4
1.2 Probleme / Verbesserungen	4
2. CodeReview C3	6
2.1 Strategie.....	6
2.2 Probleme / Verbesserungen	6

1. CodeReview C2

1.1 Strategie

Der Code wird systematisch nach Unschönheiten, zu hohen Kopplungen, unterschiedlichen Benennungen durchsucht. In der folgenden Tabelle werden die Massnamen zur Verbesserung des Codes erläutert und durchgeführt. Dies ist der erste von zwei Code Reviews. Hier werden die größten Designfehler behoben. In der nächsten Phase (CodeReview C3) sollten nur noch wenige Verbesserungen nötig sein.

1.2 Probleme / Verbesserungen

Home > Ein/Ausblenden

Klasse	View/Home
Problem	Ausblenden von Buttons per Javascript
Massnahme	Für das Anzeigen/Ausblenden von Elementen wird nur Razor eingesetzt. Durch diverse <code>@if()</code> können Elemente einfacher ein- und ausgeblendet werden als mit Javascript. Ausserdem ist der Code wesentlich kompakter.

Controller > Zu viele identische transaction() Methoden

Klasse	Alle Controller Klassen
Problem	In jeder Controller Klasse gibt es eine transaction() Methode. Diese schützt vor gleichzeitigen Zugriffen durch den gleichen User. Leider ist diese Methode in jeder Controller Klasse vorhanden.
Massnahme	Im Package BusinessLogic wird eine neue Klasse „wt4uTransaction“ erstellt. Anstatt in jedem Controller eine eigene transaction Methode aufzurufen, wird eine neue wt4uTransaction erstellt. Folgendermassen können alle schreibenden Zugriffe implementiert werden: <code>New wt4uTransaction().transact(delegate) ;</code>

Startzeiten und Endzeiten nicht einheitlich benannt

Klasse	Project, Break
Problem	Für die DateTime Objekte Start und End wurden nicht immer die gleichen Benennungen verwendet. Dies führt zu unschönem Code.
Massnahme	In allen Model-Klassen wird die Startzeit und die Endzeit einheitlich deklariert. Für diese zwei Variablen ist jeweils „End“ und „Start“ zu verwenden. Mit „refactor“ können alle Variablen gleichzeitig umbenannt werden.

ID und Id

Klasse	Alle Klassen
Problem	Für die Identifikation eines Objekts wird stets eine Id verwendet. Diese wird jedoch übers ganze Programm hinweg unterschiedlich benannt, einmal Id, dann wieder ID, was einen nicht besonders guten Eindruck hinterlässt.
Massnahme	Alles was im Namen ID enthält wird in Id umbenannt. Mit „refactor“ können alle Variablen gleichzeitig umbenannt werden.

Alte Controller/View/Model Klassen entfernen

Klasse	Alle Klassen
Problem	Mit CodeFirst und dem ASP.NET Tutorial werden sehr viele Standardklassen generiert. Diese werden allerdings nie genutzt.
Massnahme	Alle Klassen in folgenden Ordnern die keinen Zweck erfüllen oder ungenutzt sind müssen gelöscht werden: <ul style="list-style-type: none">• View• Model• Controller

Generierte Methoden entfernen

Klasse	Alle Controller Klassen
Problem	Mit dem ASP.NET Tutorial werden sehr viele Methoden in die Controller geschrieben. Diese werden für unsere Applikation nicht mehr benötigt
Massnahme	Alle Methoden in den Controllerklassen die ungenutzt sind oder keinen Zweck erfüllen sollten gelöscht werden.

Auskommentierten Code entfernen

Klasse	Alle Klassen
Problem	Aus Sicherheitsgründen oder nicht Beachtung wurden teilweise Codefragmente auskommentiert die keine Nutzung mehr haben.
Massnahme	Alle Codefragmente die auskommentiert sind analysieren und anschliessend löschen.

2. CodeReview C3

2.1 Strategie

Nun wird überprüft ob alle Coding-Style-Guideline die im Projektplan aufgeführt sind auch wirklich eingehalten wurden.

2.2 Probleme / Verbesserungen

Gross und Kleinschreibung	
Klasse	Alle Businessklassen und Controllerklassen
Problem	Gemäss Coding-Style-Guidelines folgendes festgelegt: <ul style="list-style-type: none">• Klassenvariablen grossgeschrieben• Parameter kleingeschrieben• Methoden grossgeschrieben• Lokale Variablen kleingeschrieben• Klassen grossgeschrieben Dies wurde an sehr vielen Stellen leider nicht eingehalten.
Massnahme	Bei jeder Bezeichnung bei der die Gross- und Kleinschreibung nicht gemäss Coding-Style-Guidelines umgesetzt wurde, muss der Fehler behoben werden.

.First() Aufrufe	
Klasse	Alle Businessklassen und Controllerklassen
Problem	Sind keine Elemente vorhanden wirft ein .First() Aufruf eine Exception.
Massnahme	Überprüfen, wie viele Elemente vorhanden sind. Sollten keine vorhanden sein, muss dies entsprechend abgefangen werden.