

BASTA!
NET, WINDOWS, VISUAL STUDIO

Wartbare Web-Anwendungen
mit Knockout.js und
Model-View-ViewModel (MVVM)

Malte Clasen

← → http://localhost:52803/Recipe Streuselkuchen

Knockout Editor Demo

Streuselkuchen



Foto von Malte

Zubereitung

Teig

Mehl in eine Schüssel geben und eine Kuhle hineindrücken. Zucker, Hefe und lauwarmen Soja-Reis-Drink hineingeben. Mit einem Küchentuch zudecken und 15 Minuten lang gehen lassen. Margarine zerlassen und ebenfalls in die Kuhle geben. Von innenheraus mit dem restlichen Mehl verkneten. Nochmals 15 Minuten lang zugedeckt gehen lassen. Auf einem mit Backpapier ausgelegten Blech ausrollen und wieder 15 Minuten lang gehen lassen.

Streusel

Mehl, Zucker und Zimt vermischen, Margarine zerlassen und alles verkneten. Streusel auf dem ausgerollten, gegangenen Teig **gleichmäßig** verteilen und ca. 20 Minuten bei 180°C Umluft backen.

bearbeiten

Kategorien

- [Kuchen und Torten](#)
- [Kleingebäck](#)

Zutaten

Teig

- 400 g [Mehl](#)
- 75 g [Margarine](#)
- 75 g [Zucker](#)
- 150 ml Soja-Reis-Drink
- 1 Pk [Hefe, trocken](#)
- 1 Pk Vanillezucker
- [Salz](#)

Streusel

- 300 g [Mehl](#)
- 175 g [Margarine](#)
- 300 g [Zucker](#)
- [Zimt](#)

adesso

http://localhost:52803/Recipe Streuselkuchen

Knockout Editor Demo

Streuselkuchen




Foto von Malte

Kategorien

- Kuchen und Torten
- Kleingebäck

Zutaten

Teig

400	g	Mehl
75	g	Margarine
75	g	Zucker
150	ml	Soja-Reis-Drink
1	Pk	Hefe, trocken
1	Pk	Vanillezucker
		Salz

Streusel

300	g	Mehl
175	g	Margarine
300	g	Zucker
		Zimt

Zubereitung

Teig

Mehl in eine Schüssel geben und eine Kuhle hineindrücken. Zucker, Hefe und lauwarmen Soja-Reis-Drink hineingeben. Mit einem Küchentuch zudecken und 15 Minuten lang gehen lassen. Margarine zerlassen und ebenfalls in die Kuhle geben. Von innenheraus mit dem restlichen Mehl verkneten. Nochmals 15 Minuten lang zugedeckt gehen lassen. Auf einem mit Backpapier ausgelegten Blech ausrollen und wieder 15 Minuten lang gehen lassen.

Streusel

Mehl, Zucker und Zimt vermischen, Margarine zerlassen und alles verkneten. Streusel auf dem ausgerollten, gegangenen Teig **gleichmäßig** verteilen und ca. 20 Minuten bei 180°C Umluft backen.

vorschau ↶ ↷

Acceptance Tests

Feature: RecipeEditor

In order to edit a recipe

As a registered user

I want to be able to modify all properties of a recipe

Scenario: Add Category

Given I am on the "*Streuselkuchen*" recipe page

And I am in edit mode

When I click on the empty category field

And I type "*Blechkuchen*"

Then there should be 3 categories

And the title of the third category should be "*Blechkuchen*"

<h1>Streuselkuchen</h1>

```

<div>Foto von Malte</div>
```

```
<h2>Zubereitung</h2>
```

```
<h3>Teig</h3>
```

```
<div>Mehl in eine Schüssel geben und eine
Kuhle hineindrücken. Zucker, Hefe und
<emph>lauwarmen</emph>
Soja-Reis-Drink hineingeben. Mit einem
Küchentuch zudecken und 15 Minuten lang
gehen lassen. Margarine zerlassen und
ebenfalls in die Kuhle geben. Von innenheraus
mit dem restlichen Mehl verkneten. Nochmals
15 Minuten lang zugedeckt gehen lassen. Auf
einem mit Backpapier ausgelegten Blech
ausrollen und wieder 15 Minuten lang gehen
lassen.</div>
```

```
<h3>Streusel</h3>
```

```
<div>Mehl, Zucker und Zimt vermischen,
Margarine zerlassen und alles verkneten.
Streusel auf dem ausgerollten, gegangenen
Teig <strong>gleichmäßig</strong> verteilen
und ca. 20 Minuten bei 180°C Umluft
backen.</div>
```

```
<h3>Kategorien</h3>
```

```
<ul>
  <li><a href="/KuchenUndTorten">
    Kuchen und Torten</a></li>
  <li>Kleingebäck</li>
</ul>
```

```
<h2>Zutaten</h2>
```

```
<h3>Teig</h3>
```

```
<table>
<tbody>
<tr class="ingredient">
  <td class="amount">400</td>
  <td class="unit">g</td>
  <td class="name"><a href="/Mehl">
    Mehl</a></td>
</tr>
...
</tbody>
</table>
```

```
<h3>Streusel</h3>
```

```
<table>...</table>
```

ASP.NET-Editor

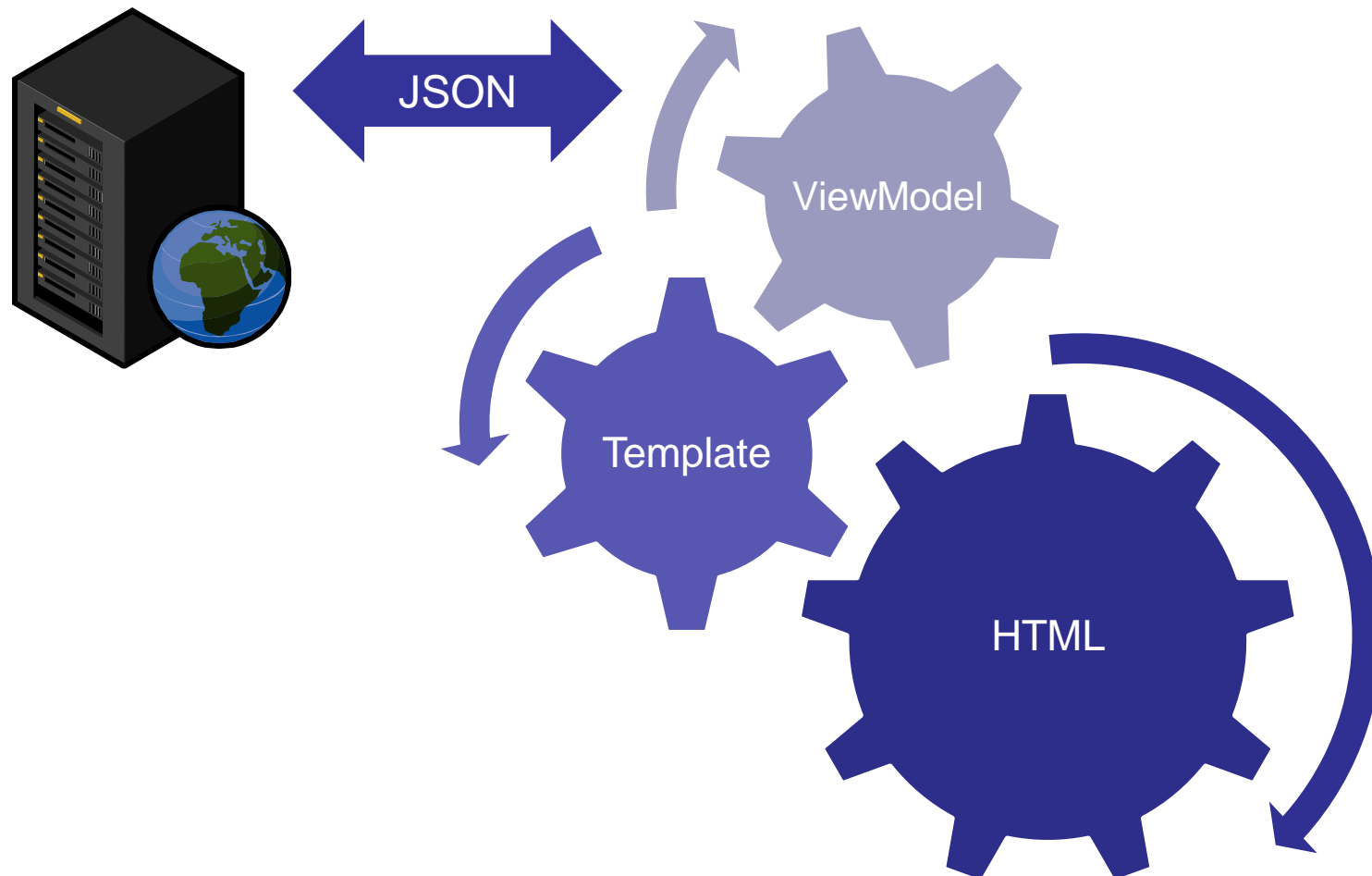
veränderbare
Anzahl in Listen

„neue
Kategorie“

zwei Code-
Pfade,
Postback, ...



Knockout: Prinzip



JSON

```
[AcceptVerbs(HttpVerbs.Get)]  
public ActionResult Get()  
{  
    return Json(_model, JsonRequestBehavior.AllowGet);  
}
```

```
[AcceptVerbs(HttpVerbs.Post)]  
public ActionResult Edit(RecipeModel model)  
{  
    return Json("gespeichert um " +  
        DateTime.Now.ToShortTimeString());  
}
```


ViewModel

```
function RecipeViewModel(initialData) {  
    var self = this;  
    self.data = ko.observable();  
    ...  
    this.onSave = function () { ... };  
    this.onRemoveComponent = function (model, event) {  
        self.data().Components.remove(model);  
    }  
    ...  
    var mapping = { ... };  
    self.data(ko.mapping.fromJS(initialData, mapping));  
};
```

Observable

```
function QuantityModel(data) {  
    this.Amount = ko.observable(data.Amount);  
    this.Unit = ko.observable(data.Unit);  
}
```

ObservableArray

```
function ComponentModel(data) {  
    var self = this;  
    this.Title = ko.observable(data.Title);  
    this.Preparation = ko.observable(data.Preparation);  
  
    this.Ingredients = ko.observableArray();  
    this.Ingredients($.map(data.Ingredients, function (item)  
        { return new RecipeIngredientModel(item); }));  
    ...  
}
```

Binding: Text

```
<h1 data-bind="text: Title"></h1>
```

ergibt zur Laufzeit

```
<h1 data-bind="text: Title">Streuselkuchen</h1>
```

Binding: With

```
function RecipeViewModel(initialData) {  
    var self = this;  
    self.data = ko.observable();  
    ...  
};
```

```
...  
ko.applyBindings(recipeViewModel);
```

```
<form action="#" id="RecipeEditor" data-bind="with: data">  
    <h1 data-bind="text: Title"></h1>  
    ...  
</form>
```

Binding: Attr

```
<ul data-bind="foreach: Categories">  
    <li><a data-bind="text:Name, attr:  
{href:Url}"></a></li>  
</ul>
```

Virtual Elements

```
<img class="photo" alt="recipe" data-bind="attr: {src:  
Image.Url}" />
```

```
<div>
```

```
    Foto von
```

```
    <!--ko text: Image.Author-->
```

```
    <!--/ko-->
```

```
</div>
```

Binding: Foreach

```
<ul data-bind="foreach: Categories">  
  <li><a data-bind="text:Name, attr: {href:Url}"></a></li>  
</ul>
```


Binding: If

```
<ul data-bind="foreach: Categories">
  <!-- ko if:!IsEmpty() -->
  <li><a data-bind="text:Name, attr: {href:Url}"></a></li>
  <!-- /ko -->
</ul>
```

```
function CategoryModel(data) {
  var self = this;
  this.Name = ko.observable(data.Name);
  this.Url = ko.observable(data.Url);
  this.IsEmpty = ko.computed(function () {
    return (!self.Name()) && (!self.Url());
  }, this);
}
```

Mapping, manuell

```
function ComponentModel(data) {  
    var self = this;  
    this.Title = ko.observable(data.Title);  
    this.Preparation = ko.observable(data.Preparation);  
    this.Ingredients = ko.observableArray();  
    this.Ingredients($.map(data.Ingredients, function (item)  
        { return new RecipeIngredientModel(item); }));  
}
```

Mapping, automatisch

```
function RecipeViewModel(initialData) {  
    var self = this;  
    self.data = ko.observable();  
    var mapping = {  
        'Components': {  
            create: function (options) {  
                return new  
                ComponentModel(options.data);  
            }  
        }  
    };  
    self.data(ko.mapping.fromJS(initialData, mapping));  
}
```

Mapping, zu JSON

```
this.onSave = function () {  
    $.ajax("/Recipe/Edit", {  
        data: ko.mapping.toJSON(self.data),  
        type: "post",  
        contentType: "application/json",  
        dataType: "json",  
        success: function (result) {  
            ShowLogMessage(result); }  
    })  
    .error(function (e, jqxhr, settings, exception) {  
        console.log(exception);  
    });  
};
```

<h1 contenteditable="true">Streuselkuchen</h1>

```

<div>Foto von Malte</div>
```

```
<h2>Zubereitung</h2>
```

```
<h3 contenteditable="true">Teig</h3>
```

```
<div contenteditable="true">Mehl in eine
Schüssel geben und eine Kuhle
hineindrücken. Zucker, Hefe und
<emph>lauwarmen</emph>
Soja-Reis-Drink hineingeben. Mit einem
Küchentuch zudecken und 15 Minuten lang
gehen lassen. Margarine zerlassen und
ebenfalls in die Kuhle geben. Von innenheraus
mit dem restlichen Mehl verkneten. Nochmals
15 Minuten lang zugedeckt gehen lassen. Auf
einem mit Backpapier ausgelegten Blech
ausrollen und wieder 15 Minuten lang gehen
lassen.</div>
```

```
<h3 contenteditable="true">Streusel</h3>
```

```
<div contenteditable="true">Mehl, Zucker
und Zimt vermischen, Margarine zerlassen
und alles verkneten. Streusel auf dem
ausgerollten, gegangenen Teig
<strong>gleichmäßig</strong> verteilen und
ca. 20 Minuten bei 180°C Umluft
backen.</div>
```

```
<h3>Kategorien</h3>
```

```
<ul>
  <li><input type="text"></li>
  <li><input type="text"></li>
</ul>
```

```
<h2>Zutaten</h2>
```

```
<div class="toolbar edit">
  <button type="button">
    
  </button>
</div>
```

```
<h3 contenteditable="true">Teig</h3>
```

```
<table>
<tbody>
<tr class="ingredient">
  <td class="amount"><input
type="number"></td>
  <td class="unit"><input
type="text"></td></td>
  <td class="name"><input
type="text"></td>
</tr>
```

```
</tr>
```

```
...
```

```
</tbody>
</table>
```

```
<h3 contenteditable="true">Streusel</h3>
<table>...</table>
```

HTML-Editor

```
<div contenteditable="true" data-bind="html: Preparation,  
event: {change: onUpdatePreparation}">Mehl in eine Schüssel  
geben ...
```

```
function ComponentModel(data) {  
    var self = this;  
    this.Preparation = ko.observable(data.Preparation);  
    this.onUpdatePreparation = function(model, event) {  
        self.Preparation($(event.target).html());  
    };  
}
```

Event: Click

```
<button type="button" data-bind="event: { click:  
$root.onRemoveComponent }">  
      
</button>
```

```
function RecipeViewModel(initialData) {  
    var self = this;  
    self.data = ko.observable();  
    this.onRemoveComponent = function (model, event) {  
        self.data().Components.remove(model);  
    }  
}
```

Binding: Value

```
<ul data-bind="foreach: Categories">  
  <li><input type="text" data-bind="value:Name"/></li>  
</ul>
```

```
function CategoryModel(data) {  
  var self = this;  
  this.Name = ko.observable(data.Name);  
}
```


Event: Change

```
<ul data-bind="foreach: Categories">  
  <li><input type="text" data-bind="value:Name,  
event:{change: $root.onChangeCategory}" /></li>  
</ul>
```

```
this.onChangeCategory = function (model, event) {  
  if (model.Name()) {  
    ...  
  } else {  
    self.data().Categories.remove(model);  
  }  
}
```

Computed

```
function ComponentModel(data) {  
    var self = this;  
    this.Ingredients = ko.observableArray();  
    this.LastIngredient = ko.computed(function () {  
        if (self.Ingredients().length === 0)  
            return null;  
        return  
self.Ingredients()[self.Ingredients().length-1];  
    }, this);  
}
```

Subscribe

```
this.LastIngredientSubscription = null;
this.LastIngredient.subscribe(function (newValue) {
    if (self.LastIngredientSubscription !== null) {
        self.LastIngredientSubscription.dispose();
    }
    var currentLastIngredient = newValue;
    if (currentLastIngredient === null)
        currentLastIngredient = self.AddEmptyIngredient();
    if (!currentLastIngredient.IsEmpty())
        currentLastIngredient = self.AddEmptyIngredient();
    self.LastIngredientSubscription =
currentLastIngredient.IsEmpty.subscribe(
        function (newValue) { if (!newValue) {
self.AddEmptyIngredient(); } });
});
```

Rahmenwerk

```
@Scripts.Render("~/Scripts/recipe")
<script type="text/javascript">
    InitRecipeViewModel(@Html.Raw(Model.Json));
</script>

var recipeViewModel;
function InitRecipeViewModel(initialData) {
    recipeViewModel = new RecipeViewModel(initialData);
}
function InitRecipeEditor() {
    ko.applyBindings(recipeViewModel);
    ShowRecipeEditor();
}
```

Übersicht

- Grundlage: HTML-Ansicht
- Knockout:
 - Model: Service
 - View: Template
 - ViewModel: Script
- Observable, ObservableArray, Compute, Subscribe
- Binding
 - Inhalte: Text, Attr
 - Struktur: With, Foreach, If
 - Events: Click

Vielen Dank für Ihre
Aufmerksamkeit.

info@adesso.de

www.adesso.de | malteclasen.de/blog