

---

# Vorwort zur 3. Auflage

Liebe Leserinnen und Leser,

herzlichen willkommen zur dritten, erheblich erweiterten Auflage dieses Buchs.

## Warum dieses Buch?

Das Web hat sich verändert. Während für gute Webanwendungen vor rund zehn Jahren noch die Regel galt, so viele Aufgaben wie möglich auf dem Server zu erledigen (Server Side Rendering), wird heutzutage in vielen Fällen die Benutzeroberfläche auf dem Client gerendert (Client Side Rendering – CSR), sodass eine Single Page Application (SPA) entsteht. Der Webserver kommt oft nur noch für das initiale Laden der SPA und zum Datenaustausch via REST-Diensten (WebAPIs) zum Einsatz. Dank Basistechniken wie HTML5/CSS3, der erweiterten JavaScript-Syntax, und vielen neuen Browser-APIs sowie zahlreichen Frameworks, die die Handhabung einfacher machen, sind heutzutage Webanwendungen möglich, die in ihrer Benutzerfreundlichkeit Desktop-Anwendungen nahekommen.

Mittlerweile werden mit Webtechniken nicht nur Browseranwendungen, sondern auch mobile Apps und Desktop-Anwendungen entwickelt. Die Produktkombination aus HTML, CSS und JavaScript ist die einzige echte Cross-Platform-Technik, die die Entwicklung von Oberflächen für alle Betriebssysteme im Sinne von »Write once, run anywhere« ermöglicht.

Trotz des Trends zur Verlagerung des Renderings auf den Client gibt es weiterhin viele Webanwendungen, bei denen mit Server Side Rendering (SSR) gearbeitet wird.

# Was ist der Inhalt dieses Buchs?

Dieses Buch beschreibt sowohl Server- als auch Clienttechniken für moderne Webanwendungen.

Anders als die Vorlagen beginnt das Buch mit einem Buchteil »Web-Basiswissen«, in dem HTTP, HTML und CSS besprochen werden. Diese Ergänzung findet bewusst statt, weil es Leser-Feedback zu den Voraufgaben gab, dass das Buch Vorwissen in diesen Bereichen voraussetzte. Natürlich kann hier nicht das komplette Wissen zu den Basistechniken des Webs abgedruckt werden. Die Basiskapitel von Jörg Krause geben Ihnen aber ein solides Fundament für den Rest des Buchs.

Serverseitig gab es im Jahr 2016 mit dem Erscheinen von ASP.NET Core einen erheblichen Umbruch, den allerdings viele Unternehmen technisch und/oder organisatorisch noch nicht nachvollziehen können. In diesem Buch wird daher auf der Serverseite sowohl das klassische ASP.NET (MVC, Web API, SignalR) im Buchteil B als auch das neue ASP.NET Core (inkl. MVC, Razor Pages und Web API, SignalR) im Buchteil C besprochen. Ein eigenes Kapitel widmet sich der Entwicklung von Microservices inklusive einer umfangreichen Fallstudie.

Im Buchteil D geht es dann um die Client-Programmierung mit JavaScript und TypeScript sowie die derzeit sehr weit verbreiteten Webframeworks Bootstrap (von Twitter) und Angular (von Google). Informationen zu React (von Facebook) finden Sie in Kapitel 30, einem der drei Bonuskapitel, die Sie als Ergänzung des Buchs – so wie im Vorwort beschrieben – herunterladen können.

Ein ganz aktuelles Kapitel zu ASP.NET Blazor ergänzt diesen Buchteil. Auch wenn ASP.NET Blazor noch in einem früheren Entwicklungsstadium steckt, weckt es dennoch bei vielen .NET-Entwicklern die Hoffnung, zukünftig mit C# statt mit JavaScript/TypeScript SPAs im Browser programmieren zu können.

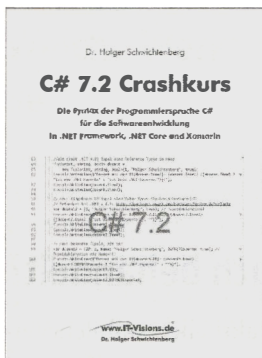
Buchteil E behandelt den Betrieb (das »Hosting«) von Webanwendungen in den Internet Information Services (IIS) von Windows, in Docker-Containern und in Microsoft Azure.

Das Buch beschließt den Buchteil F mit einem größeren Fallbeispiel: MiracleList ist eine moderne, auf Webtechniken basierende Cross-Platform-Anwendung zur Aufgabenverwaltung (in Anlehnung an *www.wunderlist.com*, für das Microsoft im Jahr 2015 einen dreistelligen Millionenbetrag bezahlt hat). Der Server basiert auf ASP.NET Core und läuft auf Windows, Linux oder macOS. Der Client basiert unter anderem auf HTML, CSS, JavaScript, Angular, Electron und Cordova. Er läuft daher sowohl im Browser als auch in Desktop-Anwendungen auf Windows, Linux oder macOS sowie als mobile App in iOS, Android und Windows Mobile.

# Welche Voraussetzungen sollten Sie für dieses Buch mitbringen?

Das Buch richtet sich an Softwareentwickler, die bereits grundlegende Erfahrung mit .NET und C# besitzen und nun moderne Webanwendungen entwickeln wollen. Das Buch beinhaltet keine Einführung in .NET und C#, aber es führt in alle notwendigen Webtechniken ein.

Wenn Sie die Programmiersprache C# noch nicht oder noch nicht gut beherrschen, sollten Sie vorab dieses Buch lesen:



Holger Schwichtenberg:

C# 7.2 Crashkurs: Die Syntax der Programmiersprache C# für die Softwareentwicklung in .NET Framework, .NET Core und Xamarin.

[www.IT-Visions.de](http://www.IT-Visions.de), 2018

1. Auflage 2018  
167 Seiten, 15,00 €

ISBN: 978-3-93427-930-8

## Was ist das Ziel des Buchs?

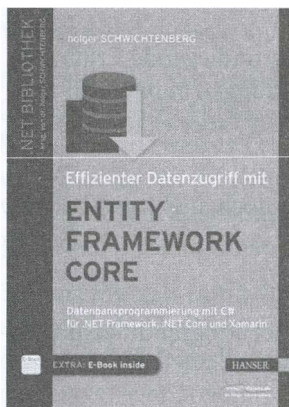
Zielsetzung dieses Buchs ist es, dem Leser zu zeigen, wie die Techniken eingesetzt werden können, um moderne Webanwendungen zu schaffen. Der Fokus liegt dabei auf den Konzepten, die hinter diesen Techniken stehen. Dazu spielt dieses Buch eine Vielzahl an Beispielen durch.

Es ist nicht Ziel dieses Buchs, ein Nachschlagewerk für die Vielzahl der Sprach- und Syntaxelemente sowie Klassen zu sein. Ein Nachschlagewerk zu all diesen Techniken würde viele Tausend Seiten dick sein. Schon bei der hier vorliegenden Seitenzahl mussten Inhalte gekürzt werden. Auf einige Fakten und Praxisbeispiele, die wir gerne präsentiert hätten, mussten wir daher leider verzichten.

Für eine erschöpfende Auflistung sämtlicher Befehle wird, gerade bei den clientseitigen Technologien, auf die umfangreichen frei verfügbaren Online-Referenzen verwiesen.

Darüber hinaus gehen wir auch davon aus, dass der Leser mit wenig Aufwand herausfinden kann, in welchem Namensraum beziehungsweise in welcher Assembly sich eine erwähnte Klasse befindet, sofern diese nicht ohnehin durch die in Visual Studio verwendeten Vorlagen eingebunden wurde.

Außerdem werden in diesem Buch keine Datenbankzugriffe besprochen. Hierzu sei auf das folgende Buch verwiesen:



Holger Schwichtenberg:

Effizienter Datenzugriff mit Entity Framework Core:  
Datenbankprogrammierung mit C# für .NET Framework,  
.NET Core und Xamarin.

Carl Hanser Verlag, 2018

468 Seiten, 42,00 €

ISBN: 978-3-446-44898-8

## Welche Versionen werden in diesem Buch behandelt?

Das Buch behandelt die zum Redaktionsschluss des Buchs aktuellen stabilen Versionen. Dies sind unter anderem:

- Visual Studio 2017 Version 15.6 und 15.7
- Visual Studio Code 1.22
- .NET Framework 4.7.2
- .NET Core 2.0.2 (mit Ausblick auf die Version 2.1)
- ASP.NET MVC 5.2.4
- ASP.NET Web API 5.2.4
- ASP.NET Core 2.0.2 (mit Ausblick auf die Version 2.1)
- Bootstrap 4.0.0
- TypeScript 2.8.1
- Angular 5.2.9
- React 16.2.0
- Internet Information Services 10.0
- Docker 18.05.0
- Electron 1.8.3
- Cordova 8.0

## Welche Programmiersprache wird in diesem Buch verwendet?

Das vorliegende Buch verwendet serverseitig die Programmiersprache C# sowie clientseitig JavaScript und TypeScript.

Alle Leser, die serverseitig lieber mit Visual Basic .NET arbeiten, können die abgedruckten Beispiele sehr einfach mit kostenlosen Werkzeugen nach Visual Basic .NET konvertieren. Informationen dazu findet man unter <http://www.dotnetframework.de/tools.aspx>.

## Welche Werkzeuge werden in diesem Buch verwendet?

Dieses Buch bezieht sich primär auf die Programmierung mit der integrierten Entwicklungsumgebung Visual Studio. Von Visual Studio kann man eine kostenfreie Community Edition nur in kleineren Firmen und bei Open-Source-Projekten kostenfrei einsetzen ([www.visualstudio.com/vs/community](http://www.visualstudio.com/vs/community)). Für die Webclient-Programmierung kommt auch Visual Studio Code (<https://code.visualstudio.com>) zum Einsatz. Zudem sollten Sie node.js ([nodejs.org](http://nodejs.org)) installieren, um den Node Package Manager (NPM) zur Verfügung zu haben. Für das Testen von Web APIs wird Postman ([www.getpostman.com](http://www.getpostman.com)) und Fiddler ([www.telerik.com/download/fiddler](http://www.telerik.com/download/fiddler)) verwendet. Auf weitere Werkzeuge wird in den einzelnen Kapiteln hingewiesen.

## Welche Sprachversion wird in diesem Buch verwendet?

Dieses Buch beschreibt die englische Version von Visual Studio, weil inzwischen viele deutsche Entwickler (einschließlich der Autoren) die englische Version der Software bevorzugen, denn die Übersetzungen ins Deutsche sind oft holprig und die Fehlermeldungen nur schwer verständlich. Visual-Studio-Anwender können über den Befehl *Tools/Options/Environment/International Settings* weitere Sprachpakete herunterladen und einrichten. Weiterhin sei noch darauf hingewiesen, dass die Anordnung der Menüs und auch einige Tastaturkürzel von den gewählten Einstellungen in Visual Studio abhängen. Alle Ausführungen in diesem Buch beziehen sich auf die Umgebungseinstellung *Common Settings*, die bei der Installation des Produkts ausgewählt werden kann.

# Woher bekommen Sie die Code-Beispiele und die drei Bonuskapitel zu diesem Buch?



Die Code-Beispiele zu diesem Buch erhalten Sie auf [oreilly.de](http://oreilly.de) auf der Webseite zum Buch *Moderne Webanwendungen für .NET-Entwickler* unter »Zusatzmaterial«.

Hier finden Sie außerdem drei zusätzliche Kapitel als PDF zum Download, die dieses Buch inhaltlich abrunden: Kapitel 28 »Open Web Interface for .NET (OWIN) und Katana«, Kapitel 29 »Sicherheit in ASP.NET« sowie Kapitel 30 »React«.

Die Autoren bieten den Leserinnen und Lesern dieses Buchs darüber hinaus im Rahmen einer zugangsbeschränkten Website folgende Serviceleistungen:

- **Bonuskapitel** Ergänzend erhalten Sie auch hier die drei zusätzlichen Kapitel als PDF zum Download: Kapitel 28 »Open Web Interface for .NET (OWIN) und Katana«, Kapitel 29 »Sicherheit in ASP.NET« sowie Kapitel 30 »React«.
- **Downloads** Sie können alle in diesem Buch vorgestellten Code-Beispiele hier herunterladen.
- **Spickzettel (Cheat Sheets)** Ergänzend erhalten Sie hier drei Spickzettel in PDF-Form mit den wichtigsten Befehlen von ASP.NET (Core) MVC und Web API, TypeScript und Angular – zum Ausdrucken und »Spicken« beim Programmieren.
- **Diskussionsrunde** Ein webbasiertes Forum bietet die Möglichkeit, Fragen an die Autoren zu stellen. Bitte beachten Sie jedoch, dass dies eine freiwillige Leistung der Autoren ist und kein Anspruch auf eine kostenlose Betreuung besteht.
- **Newsletter** Alle registrierten Leser erhalten zwei- bis viermal jährlich einen Newsletter mit aktuellen Terminen und Publikationshinweisen.
- **Leserbewertung** Vergeben Sie Noten für dieses Buch und lesen Sie nach, was andere Leser von diesem Buch halten.
- **Errata** Trotz eines jahrelang erprobten Vorgehensmodells und der dreifachen Qualitätskontrolle (Co-Autor, Fachlektor, Verlag) ist es möglich, dass sich einzelne Fehler in dieses Buch eingeschlichen haben. Im Webportal können Sie nachlesen, welche Fehler gefunden wurden. Sie können hier auch selbst Fehler, die Ihnen auffallen, melden.

## Spickzettel („Cheat Sheet“) ASP.NET (Core) MVC & WebAPI

Autor: Dr. Holger Schwichtenberg (www.IT-Visions.de)

V1.02 (PDF) / 04.06.2017 / Seite 1 von 2



### ASP.NET MVC:

```
using System.Web;
using System.Web.Mvc;
```

### ASP.NET Core MVC:

```
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.ViewEngines;
using Microsoft.AspNetCore.Mvc.ViewFeatures;
```

```
public void ConfigureServices(IApplicationBuilder app, IHostingEnvironment
    env, ILoggerFactory loggerFactory)
{
    // Injektion der Controller
    app.UseMvc(routes => {
        app.UseDefaultMvcOptions(new
            {
                app.UseErrorPage(LoadedViewHomeError);
            });
        app.UseDefaultFiles();
        app.UseStaticFiles();
        app.UseSession();
        app.UseRouting();
        app.UseAuthentication();
        app.UseAuthorization();
    });
}
```

```
public class HomeController
{
    // Aufruf = Standardansicht / Index
    public string Index()
    {
        return System.DateTime.Now.ToString("dd.MM.yyyy");
    }
    // Aufruf = Standardansicht / View/Action/Index
    public ActionResult<ViewResult> Index()
    {
        return new ContentResult {
            ContentType = "text/html",
            Content = "<h1>=>body</h1>=>Mein Name ist " + name
                + "</h2>=> </body>=> </html>";
        };
    }
}
```

### Controller mit View

```
public class FlightController : Controller
{
    // verwendet einen Entity Framework-basierten Kontext
}
```

```
private EFContext ctx = new EFContext();
public void Display() => ctx.Bruecke();
// Aufruf = Standardroute // Flag
public ActionResult Index()
{
    ViewBag.GesamtAnzahl = ctx.Flight.Count();
    var Flug = ctx.Flight.OrderBy(f => f.Datum).Take(10);
    return View(Flug); // zeigt View/Bruecke/Index.chtml
}
```

```
// Aufruf = Standardroute // Flag/Detail/123
public ActionResult Details(int id)
{
    var Flug = ctx.Flight.Find(id); // Sucht per PK
    return View(Flug); // zeigt View/Bruecke/Detail.chtml
}
```

```
// Aufruf // Flag/Routen/10/20
[Route("Flug/{abflugort}")]
public ActionResult Index(string abflugort, int von, int bis)
{
    ViewBag.GesamtAnzahl = ctx.Flight.Count();
    var Flug = ctx.Flight.Where(f => f.Abflugort == abflugort)
        .OrderBy(f => f.Datum).Take(10);
    return View(Model, ViewBag); // zeigt View/Bruecke/Index.chtml
}
```

```
// Aufruf: http://meinServer/Flyer/Controller/Index/10/1/1
[Route("Flug/Flieger/{id}/{anzahl}")]
public ActionResult<ViewResult> Index(int id, int anzahl)
{
    var Flug = ctx.Flight.OrderBy(f => f.Datum).Take(10);
    Flug.Flyer = anzahl;
    return View(Flug); // zeigt View/Bruecke/Index.chtml
}
```

```
// Formularerkennung mit POST
[HttpPost]
public ActionResult<ViewResult> Index(Flight Flug)
{
    // Standardroute // Flag
}
```

```
// Standardroute // Flag
[Route("Flug/{id}")]
public ActionResult Index(int id)
{
    return Redirect("Account/Logout");
}
```

```
}
}
```

### WebAPI-Controller

```
[Route("api/{controller}")]
// bei ASP.NET Core: Basisklasse nur "Controller"
public class FlightApiController : ApiController
{
    private FlightContext ctx = new FlightContext();
    // Aufruf /api/Bruecke/123
    [HttpGet]
    public Flight Get(int id)
    {
        return ctx.Flight.Find(id);
    }
    // Aufruf /api/Bruecke/123/{id}
    [HttpGet]
    public Flight Get(int id, int start)
    {
        return ctx.Flight.Skip(start).Take(10);
    }
}
```

```
using CO
[Route("api/Bruecke/CollectionView")]
public ActionResult<PageResult> Flighter(int br-
    uecke, int start, int end)
{
    return ctx.Flight.Skip(start).Take(end-start+1);
}
```

```
using CO
[Route("api/Bruecke/CollectionView")]
public ActionResult<PageResult> Flighter(int br-
    uecke, int start, int end)
{
    return ctx.Flight.Skip(start).Take(end-start+1);
}
```

### Angaben vor allen Views mit ViewStart.chtml

```
<!-- Layout = "_Layout" -->
@using CO
@using System.Globalization
@addTagHelper "*, Microsoft.AspNetCore.Mvc.TagHelpers"
```

### Razor-Ausdrücke

```
Ausgabe ist HTML Encoded
@asp_encode
Ausgabe ist nicht Encoded
@raw_html
Durch das Leerzeichen ist der Punkt reiner Text.
Dies ist die @: Seite!
```

Abbildung V.1: Der Spickzettel zu ASP.NET (Core) MVC und Web API

Zugang zu der ehrenamtlich betriebenen Leser-Website erhalten Sie unter [www.IT-Visions.de/Leser](http://www.IT-Visions.de/Leser). Dort müssen Sie sich einmalig registrieren. Bei der Registrierung wird ein **Losungswort** abgefragt, das Sie als Käufer dieses Buchs ausweist. Bitte geben Sie dort **Beyond** ein. Durch die Registrierung erhalten Sie ein persönliches Kennwort per E-Mail zugesendet, das Sie dann für die Anmeldung nutzen können.



Dieses Symbol weist auf eine Warnung oder ein Problem hin.



Dieses Symbol markiert einen allgemeinen Hinweis oder Tipp.

# Wem ist zu danken?

Für ihre Mitwirkung an diesem Buch möchte ich danken:

- meinem Co-Autor Jörg Krause, der den Großteil des Texts im Buchteil »Web-Basiswissen« sowie die Kapitel zu JavaScript, Bootstrap und React geschrieben hat.
- meinem ehemaligen Co-Autor Manfred Steyer, der sich nun leider nicht mehr mit den Themen dieses Buchs beschäftigt, von dem wir aber einige Texte zum klassischen ASP.NET MVC und Web API in die neue Auflage übernehmen durften.
- Dr. Joachim Fuchs für das Kapitel »ASP.NET Core SignalR«.
- Sebastian Kleinschmager für die Kapitel zu ASP.NET Core Web APIs und Microservices.
- der O'Reilly-Lektorin Ariane Hesse.
- der Korrektorin Claudia Lötschert, die das Buch sprachlich verbessert hat.

Essen im Herzen des Ruhrgebiets, im April 2018

*Holger Schwichtenberg*

## Die Autoren

### Über [www.IT-Visions.de](http://www.IT-Visions.de)



Alle Autoren dieses Buchs arbeiten bei der Firma [www.IT-Visions.de](http://www.IT-Visions.de) als Softwarearchitekten, Softwareentwickler, Trainer und Berater für .NET-Techniken. [www.IT-Visions.de](http://www.IT-Visions.de) ist ein Verbund der deutschen Top-Experten im Bereich der Microsoft-Produkte und -Technologien insbesondere .NET. Unter Leitung und Mitwirkung von Dr. Holger Schwichtenberg bietet [www.IT-Visions.de](http://www.IT-Visions.de):

- Strategische und technische Beratung
- Konzepte, Machbarkeitsstudien und Reviews
- Coaching bei Entwicklungsprojekten
- Technischen Support vor Ort und via Telefon, E-Mail oder Web-Konferenz
- Individuell zugeschnittene technische Vor-Ort-Schulungen und anforderungsorientierte Workshops
- Öffentliche Seminare, siehe [www.dotnet-akademie.de](http://www.dotnet-akademie.de)



Zu den Kunden gehören neben vielen mittelständischen Unternehmen auch Großunternehmen wie z.B. Siemens, Bayer, RWE, Bertelsmann, BASF, Bosch, Thyssen-Krupp, Merkle, Festo, Lufthansa, AIRBUS, REWE, EnBW, DEA, PwC, Deutsche Telekom, Deutsche Post, Karstadt, eurofins, Roche, OSRAM, Jenoptik, Continental, Carl Zeiss, Haufe-Lexware, TÜV, Sartorius, Fraunhofer, diverse Banken und Versicherungen sowie mehrere Landesregierungen.

- Firmenwebsites: <http://www.IT-Visions.de> und <http://www.5minds.de>
- Kontakt zu allen Autoren: E-Mail **buero@IT-Visions.de** sowie Telefon **0201-64 95 90-0**

### Dr. Holger Schwichtenberg

- Studienabschluss Diplom-Wirtschaftsinformatik an der Universität Essen
- Promotion an der Universität Essen im Gebiet komponentenbasierter Softwareentwicklung
- Seit 1996 selbstständig als unabhängiger Berater, Dozent, Softwarearchitekt und Fachjournalist
- Leiter des Berater- und Dozententeams bei [www.IT-Visions.de](http://www.IT-Visions.de)
- Softwareentwicklungsprojektleitung im Bereich Microsoft/.NET bei der *5minds IT-Solutions GmbH & Co. KG* ([www.5minds.de](http://www.5minds.de))
- Über 65 Fachbücher beim Carl Hanser Verlag, bei O'Reilly, Microsoft Press und Addison-Wesley sowie mehr als 1000 Beiträge in Fachzeitschriften
- Gutachter in den Wettbewerbsverfahren der EU gegen Microsoft (2006–2009)
- Ständiger Mitarbeiter der Zeitschriften *iX* (seit 1999), *dotnetpro* (seit 2000) und *Windows Developer* (seit 2010) sowie beim Onlineportal [heise.de](http://heise.de) (seit 2008)
- Regelmäßiger Sprecher auf nationalen und internationalen Fachkonferenzen (zum Beispiel Microsoft TechEd, Microsoft Summit, Microsoft IT Forum, BASTA, BASTA-on-Tour, .NET Architecture Camp, Advanced Developers Conference, Developer Week, OOP, DOTNET Cologne, MD DevDays, Community in Motion, DOTNET-Konferenz, VS One, NRW.Conf, Net.Object Days, Windows Forum, Container Conf)



- Zertifikate und Auszeichnungen von Microsoft:
  - Microsoft Most Valuable Professional (MVP)
  - Microsoft Certified Solution Developer (MCSRD)
- Thematische Schwerpunkte:
  - Softwarearchitektur, mehrschichtige Softwareentwicklung, Softwarekomponenten, SOA
  - Microsoft .NET Framework, Visual Studio, C#, Visual Basic
  - .NET-Architektur/Auswahl von .NET-Technologien
  - Einführung von .NET Framework und Visual Studio/Migration auf .NET
  - Webanwendungsentwicklung und Cross-Plattform-Anwendungen mit HTML, ASP.NET, JavaScript/TypeScript und Webframeworks wie Angular
  - Enterprise .NET, verteilte Systeme/Webservices mit .NET, insbesondere Windows Communication Foundation und Web API
  - Relationale Datenbanken, XML, Datenzugriffsstrategien
  - Objektrelationales Mapping (ORM), insbesondere ADO.NET Entity Framework und EF Core
  - Windows PowerShell, PowerShell Core und Windows Management Instrumentation (WMI)

## Jörg Krause

Jörg Krause arbeitet seit vielen Jahren als Autor, Trainer, Berater und Softwareentwickler für große Unternehmen weltweit. Bauen Sie auf die Erfahrung aus mehr als 25 Jahren Arbeit mit Webumgebungen und sehr vielen großen und kleinen Projekten. Schwerpunkte der Arbeit sind Webtechnologien und damit verwandte Bereiche. Dies umfasste sowohl den Frontend- als auch den Backend-Bereich sowie Nutzung von Windows und auch von Linux mit der Wahl der jeweils optimalen Plattform.

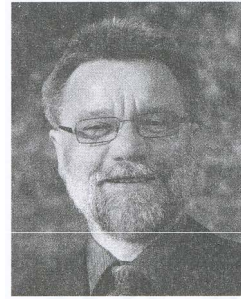


Er hat über 40 Titel bei renommierten Fachverlagen in Deutsch und Englisch veröffentlicht, darunter auch einige Bestseller. Weitere Titel sind im Eigenverlag erschienen, ebenso wie zahlreiche Zeitschriftenartikel. Sein Wissen gibt er in Seminaren und Schulungen und gelegentlich auf Fachkonferenzen weiter. Zu den Grundlagenthemen gehören HTML, CSS, JavaScript und TypeScript, aktuelle Anwendungs-Rahmenwerke wie React und Angular sowie Backend-Technologien wie beispielsweise SQL Server, Node.js, ASP.NET und Docker.

Jörg sind vor allem solide Grundlagen wichtig. Statt immer dem neuesten Framework hinterherzurrennen wären viele Entwickler besser beraten, sich eine robuste Grundlage zu schaffen. Wer dies kompakt und schnell lernen will, ist bei ihm richtig. Deshalb wird der didaktischen Aufbereitung in seinen Schulungen und Veröffentlichungen besondere Aufmerksamkeit zuteil.

## Dr. Joachim Fuchs

Dr.-Ing. Joachim Fuchs ist ein bekannter Experte für .NET mit Schwerpunkt auf UI-Techniken (insbesondere WPF, ASP.NET MVC und Angular). Er arbeitet für [www.IT-Visions.de](http://www.IT-Visions.de) als Trainer und Softwareentwickler. Durch seine Fachbücher und Artikel in Fachzeitschriften ist er in der .NET-Szene bestens bekannt.



## Sebastian Kleinschmager

Sebastian Kleinschmager ist freiberuflicher Softwareentwickler und Trainer. Seine technologischen Schwerpunkte liegen im Bereich .NET/C# und Webtechnologien, vor allem ASP.NET Core und Angular. Zudem beschäftigt er sich stark mit den Themen Clean Code, Qualitätssicherung und Architektur, dort insbesondere Microservices und ähnlichen Ansätzen. Sein Wissen zu diesen Themen bringt er mit viel Leidenschaft bei seinen Kunden in Projekte ein und vermittelt es genauso motiviert als Dozent und Teil des [www.IT-Visions.de](http://www.IT-Visions.de)-Expertenteams.



## Manfred Steyer

Manfred Steyer ist Trainer und Berater bei [www.IT-Visions.de](http://www.IT-Visions.de) sowie verantwortlich für den Fachbereich Software Engineering der Studienrichtung IT und Wirtschaftsinformatik an der FH CAMPUS 02 in Graz.

Er schreibt für das [windows.developer magazin](http://windows.developer.magazin) (vormals [dot.net magazin](http://dot.net.magazin)) sowie Heise Developer und ist Buchautor bei O'Reilly, Microsoft Press sowie Carl Hanser. Manfred hat berufsbegleitend IT und IT-Marketing in Graz sowie Computer Science in Hagen studiert und kann auf mehr als 15 Jahre an Erfahrung in der Planung und Umsetzung von großen Applikationen zurückblicken. Er ist ausgebildeter Trainer für den Bereich der Erwachsenenbildung und spricht regelmäßig auf Fachkonferenzen.

In der Vergangenheit war Manfred Steyer mehrere Jahre für ein großes österreichisches Systemhaus tätig. In der Rolle als Bereichsleiter hat er gemeinsam mit seinem Team Geschäftsanwendungen konzipiert und umgesetzt.

