**Dokumentation  
von der Entwicklung des  
Prototyps vom Projektzeiten-Manager**

# MS Visual Studio einrichten

Der Projektzeiten-Manager wird auf Basis einer Vorlage entwickelt:

* React Core Boilerplate (TypeScript)

Auf Basis der Vorlage ist die Installation von Erweiterungen für MS Visual Studio notwendig.

## Verwendete Vorlage in MS Visual Studio

Für die Entwicklung einer prototypischen Web-App des Projektzeiten-Managers wird eine Vorlage verwendet, welche React mit TypeScript integriert. Die Erweiterung für MS Visual Studio, um diese Vorlage verwenden zu können, ist unter folgendem Link abrufbar:

* React Core Boilerplate (TypeScript)
* <https://marketplace.visualstudio.com/items?itemName=NikolayMaev.ReactCoreBoilerplate>

Diese Vorlage beinhaltet die Funktionalitäten von:

* ASP.NET Core
* HTML
* CSS
* Bootstrap
  + Als CSS-Erweiterung
* JavaScript
* TypeScript
  + Als Interpreter für das objektorientierte Programmieren von JavaScript
* React
* React-Helmet
* Redux
* WebPack
  + Als Bundler für alle CSS-Funktionalitäten
  + Als Bundler für alle JavaScript- sowie TypeScript-Funktionalitäten
  + Als Bundler für alle Grafiken
* Babel
* Docker
* Server-side rendering

## Erweiterungen von MS Visual Studio

Die Erweiterungen von MS Visual Studio sind unter folgendem Menüpunkt zu finden:

* Menüpunkt „Erweiterungen“
* Auswahl „Erweiterungen verwalten“

Für den Projektzeiten-Manager werden folgende Erweiterungen von MS Visual Studio verwendet:

* TypeScript 4.4.2 for Visual Studio

# Konfigurierung der Vorlage für den Projektzeiten-Manager

## Projekt-Hierarchie

## .NET Core Basis

* In Projekteigenschaften
* Aus Vorlage
  + Von 3.1
  + Auf 5.0

## NuGet-Pakete

NuGet-Pakete sind Funktionsbibliotheken für die Programmierung von server-seitigen Funktionslogiken der Web-App.

### Übersicht der NuGet-Pakete

Der Pfad der NuGet-Pakete ist wie folgt zu finden:

* Projekt-Root-Verzeichnis/Abhängigkeiten/Pakete

Der Installationsmanager für die NuGet-Pakete lässt sich wie folgt öffnen:

* Rechtsklick auf „Pakete“
* Auswahl „NuGet-Pakete verwalten“

### Verwendete NuGet-Pakete

NuGet-Pakete aus der Vorlage mit zusätzlicher Aktualisierung:

* Microsoft.AspNetCore.SpaServices
  + Von 3.1.5
  + Auf 3.1.19
* Microsoft.AspNetCore.SpaServices.Extension
  + Von 3.1.5
  + Auf 5.0.10
* Microsoft.Extensions.Hosting
  + Von 3.1.5
  + Auf 5.0.0
* Microsoft.VisualStudio.Azure.Containers.Tools.Targets
  + Von 1.10.9
  + Auf 1.11.1
* Microsoft.VisualStudio.Web.CodeGeneration.Design
  + Von 3.1.3
  + Auf 5.0.2
* Serilog
  + Von 2.9.0
  + Auf 2.10.0
* Serilog.AspNetCore
  + Von 3.2.0
  + Auf 4.1.0
* Serilog.Extensions.Logging
  + 3.0.1
* Serilog.Sinks.Console
  + Von 3.1.1
  + Auf 4.0.0
* Serilog.Sinks.File
  + Von 4.1.0
  + Auf 5.0.0

Zusätzlich integrierte NuGet-Pakete:

* + MS.TypeScript.MSBuild
    - 4.4.2
      * Entspricht der Version von der MS Visual Studio Erweiterung  
        „TypeScript 4.4.2 for Visual Studio“

### Durchführung der Integration

Die Integration von NuGet-Paketen wird wie folgt durchgeführt:

1. Öffnen des NuGet Installationsmanagers
   * Rechtsklick auf „Projekt-Root-Verzeichnis/Abhängigkeiten/Pakete“
   * Auswahl „NuGet-Pakete verwalten“
2. Updates auf die aktuelle Version der bereits integrieten NuGet-Pakete
3. Zusätzliche Integration von NuGet-Paketen mit aktueller Version
   * MS.TypeScript.MSBuild

## Node.js NPM-Pakete

NPM-Pakete sind Funktionsbibliotheken für die Programmierung von client-seitigen Funktionslogiken der Web-App. Die NPM-Pakete werden durch Node.js installiert und integriert.

Verwendete Node.js-Version

* Aktuelle LTS Version
  + 14.17.6
* Inkompatibilität bei aktuelle Current Version
  + 16.9.1
  + Mit Paket „node-sass“

### Übersicht der NPM-Pakete

Die Konfigurations-Datei für die NPM-Pakete ist wie folgt integriert:

* Projekt-Root-Verzeichnis/package.json

Die NPM-Pakete des Projektes befinden sich im folgenden Verzeichnis

* Projekt-Root-Verzeichnis/node\_modules
* Nach erstmaliger Integration der NPM-Pakete

Die global installierten NPM-Pakete befinden sich im folgenden Verzeichnis

* C:\Program Files\nodejs\node\_modules

### Integration der NPM-Pakete

NPM-Pakete können wie folgt in das Projekt integriert werden:

* Installation nach Konfigurations-Datei
  + Ausführen MS-VS-Konsole oder Windows-Konsole
  + Verzeichnis Projekt-Root
  + Befehl: npm install
* Update nach Konfigurations-Datei
  + devDependencies
    - npm update --save-dev
  + dependencies
    - npm update –save
* Einzelnes Packet mit automatischer Integration in package.json
  + devDependencies
    - Befehl: npm install --save-dev [PACKAGES]
    - [PACKAGES] für Liste an Paketen
    - Eingabe ohne []
  + dependencies
    - Befehl: npm install --save [PACKAGES]
    - [PACKAGES] für Liste an Paketen
    - Eingabe ohne []

Die Globale Installation von NPM-Paketen erfolgt wie folgt:

* Befehl: npm install -g [PACKAGES]
* [PACKAGES] für Liste an Paketen
* Eingabe ohne []

### Pakete für den Developer-Modus

Die Abhängigen NPM-Pakete für den Developer-Modus befinden sich in der package.json unter folgenden Punkt:

* „devDependencies“: {[PACKAGES]}
* [PACKAGES] für Liste an Paketen
* Eingabe ohne []

Folgende NPM-Pakete werden verwendet:

*ASP.NET Core*

* + - aspnet-prerendering
* Von 3.0.1
* Auf ^3.0.1

*React*

* + - react-dev-utils
* Von 10.2.1
* Auf ^11.0.4
  + - react-hot-loader
* Von 4.12.21
* Auf ^4.13.0

*Noop*

* + - node-noop
* Von 1.0.0
* Auf ^1.0.0

*RimRaf*

* + - Rimraf
* Von 3.0.2
* Auf ^3.0.2

*TypeScript*

* + - @types/history
* Von 4.7.5
* Auf ^4.7.9
  + - @types/react
* Von 16.9.34
* Auf ^16.14.15
  + - @types/react-dom
* Von 16.9.7
* Auf ^16.9.14
  + - @types/react-helmet
* Von 5.0.15
* Auf ^5.0.17
  + - @types/react-redux
* Von 7.1.8
* Auf ^7.1.18
  + - @types/react-router
* Von 5.1.7
* Auf ^5.1.16
  + - @types/react-router-dom
* Von 5.1.5
* Auf ^5.1.9
  + - @types/webpack
* Von 4.41.12
* Auf ^4.41.31
  + - @types/webpack-env
* Von 1.15.2
* Auf ^1.16.2
  + - typescript
* Von 3.8.3
* Auf ^4.4.2
  + - Entspricht der Version von der MS Visual Studio Erweiterung  
      „TypeScript 4.4.2 for Visual Studio“

*WebPack*

* General import
  + webpack
    - Von 4.43.0
    - Auf ^4.46.0
  + webpack-cli
    - Von 3.3.11
    - Auf ^3.3.12
  + webpack-dev-middleware
    - Von 3.7.2
    - Auf ^3.7.3
  + webpack-hot-middleware
    - Von 2.25.0
    - Auf ^2.25.1
  + webpack-merge
    - Von 4.2.2
    - Auf ^4.2.2
  + aspnet-webpack-react
    - Von 4.0.0
    - Auf ^4.0.0
* Plugins
  + case-sensitive-paths-webpack-plugin
    - Von 2.3.0
    - Auf ^2.4.0
  + fork-ts-checker-webpack-plugin
    - Von 4.1.3
    - Auf ^6.3.3
  + terser-webpack-plugin
    - Von 5.0.3
    - Auf ^6.0.1
* CSS files loader
  + css-loader
    - Von 3.5.3
    - Auf ^3.6.0
  + style-loader
    - Von 1.2.1
    - Auf ^1.3.0
  + cssnano
    - Von 4.1.10
    - Auf ^5.0.8
  + mini-css-extract-plugin
    - Von 0.9.0
    - Auf ^0.12.0
  + optimize-css-assets-webpack-plugin
    - Von ^6.0.1
    - Auf ^6.0.1
* SASS files loader
  + sass-loader
    - Von 8.0.2
    - Auf ^8.0.2
  + node-sass
    - Von 4.14.1
    - Auf ^4.14.1
* JavaScript + TypeScript files loader
  + babel-loader
    - Von 8.1.0
    - Auf ^8.2.2
  + babel-plugin-import
    - Von 1.13.0
    - Auf ^1.13.3
* TypeScript files loader
  + ts-loader
    - Von 7.0.3
    - Auf ^7.0.5
  + ts-nameof
    - Von 4.2.2
    - Auf ^5.0.0
  + ts-nameof-loader
    - Von 1.0.2
    - Auf ^1.0.2
* Other files
  + file-loader
    - Von 6.0.0
    - Auf ^6.2.0
* Other loader
  + url-loader
    - Von 4.1.0
    - Auf ^4.1.1
  + ignore-loader
    - Von 0.1.2
    - Auf ^0.1.2

### Pakete für den Release-Modus

Die Abhängigen NPM-Pakete für den Release-Modus befinden sich in der package.json unter folgenden Punkt:

* „dependencies“: {[PACKAGES]}
* [PACKAGES] für Liste an Paketen
* Eingabe ohne []

Folgende NPM-Pakete werden verwendet:

*General import*

* awesome-debounce-promise
  + Von 2.1.0
  + Auf ^2.1.0
* axios
  + Von 0.19.2
  + Auf 0.19.2
* domain-wait
  + Von ^1.3.0
  + Auf ^1.3.0
* formik
  + Von 2.1.4
  + Auf ^2.2.9
* history
  + Von 4.10.1
  + Auf ^4.10.1
* nval-tippy
  + Von ^1.0.40
  + Auf ^1.0.40
* query-string
  + Von 6.12.1
  + Auf ^7.0.1

*Polyfill*

* custom-event-polyfill
  + Von 1.0.7
  + Auf ^1.0.7
* event-source-polyfill
  + Von 1.0.12
  + Auf ^1.0.25

*Sass*

* Sass
  + Von 1.26.5
  + Auf ^1.41.1

*JavaScript*

* core-js
  + Von ^3.6.5
  + Auf ^3.18.0
* serialize-javascript
  + Von ^4.0.0
  + Auf ^6.0.0

*Bootstrap*

* Bootstrap
  + Von ^4.4.1
  + Auf ^5.1.1

*React*

* react
  + Von 16.13.1
  + Auf ^17.0.2
* react-bootstrap
  + Von 1.0.1
  + Auf ^1.6.3
* react-dom
  + Von 16.13.1
  + Auf ^17.0.2
* react-helmet
  + Von 6.0.0
  + Auf ^6.1.0
* react-paginating
  + Von 1.4.0
  + Auf ^1.4.0
* react-redux
  + Von 7.2.0
  + Auf ^7.2.5
* react-router
  + Von 5.1.2
  + Auf ^5.2.1
* react-router-bootstrap
  + Von 0.25.0
  + Auf ^0.25.0
* react-router-dom
  + Von 5.1.2
  + Auf ^5.3.0
* react-toastify
  + Von 5.5.0
  + Auf ^8.0.2
* connected-react-router
  + Von 6.8.0
  + Auf ^6.9.1

*Redux*

* @reduxjs/toolkit
  + Von 1.3.6
  + Auf ^1.6.1
* redux
  + Von 4.0.5
  + Auf ^4.1.1
* redux-thunk
  + Von 2.3.0
  + Auf ^2.3.0

*WebPack*

* aspnet-webpack
  + Von 3.0.0
  + Auf ^3.0.0

### Durchführung der Integration

Die Integration von NPM-Paketen wird wie folgt durchgeführt:

1. Ausführen einer MS-VS-Konsole oder einer Windows-Konsole
   * Im Projekt-Root-Verzeichnis
2. Integration der NPM-Pakete
   * Befehl: npm install
3. Manuelles Prüfen der Machbarkeit von Updates bei den NPM-Paketen
   * Anpassung der Versionsnummer in der package.json
     + Anzeigen der Versionsnummer-Vorschläge
     + Tastenkombination „STRG + Leertaste“ im Feld der Versionsnummer
   * Inkompatibilität bei einigen NPM-Paket-Updates erfordert das Zurücksetzen auf eine kompatible Version
4. Updates für NPM-Pakete
   * Update der Pakete in „devDependencies“
     + Befehl: npm update --save-dev
   * Update der Pakete in „dependencies“
     + Befehl: npm update –save
5. Testen der NPM-Paket Kompatibilität
   * Durch Ausführen der Web-App
     + IIS-Express
   * Durch Ausführen des WebPack Bundlings
     + Befehl: npm run build:dev

# WebPack Bundling

# Project Build