



**Oldenburgische**  
Industrie- und Handelskammer

Abschlussprüfung Sommer 2025

Fachinformatiker für Anwendungsentwicklung

**Dokumentation zur betrieblichen Projektarbeit**

# **Formular-Editor Schnittstelle**

**Implementierung eines dynamischen Formular-Editors  
für die GSCS der Green Solutions Software GmbH**

Abgabedatum: Oldenburg, den 10.05.2025

**Prüfungsbewerber:**

Mustafa Shahin  
Grenadierweg 2  
26129 Oldenburg

**Ausbildungsbetrieb:**

Green Solutions Software GmbH  
Eva-Lessing-Straße 6  
26160 Bad Zwischenahn



**green solutions**  
cloud software

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
1.1	Projektumfeld . . . . .	2
1.2	Projektziel . . . . .	2
1.3	Projektbegründung . . . . .	2
1.4	Projektschnittstellen . . . . .	3
1.5	Projektabgrenzung . . . . .	3
<b>2</b>	<b>Planung</b>	<b>3</b>
2.1	Projektphasen . . . . .	3
2.2	Ressourcenplanung . . . . .	4
2.3	Entwicklungsprozess . . . . .	4
<b>3</b>	<b>Analyse</b>	<b>4</b>
3.1	Durchführung Ist-Analyse . . . . .	4
3.2	Durchführung Soll-Analyse . . . . .	4
3.3	Durchführung der Wirtschaftlichkeitsanalyse und Amortisationsrechnung . . . . .	5
<b>4</b>	<b>Entwurf</b>	<b>5</b>
4.1	Entwurf der Benutzeroberfläche . . . . .	5
4.2	Entwurf der Komponentenstruktur . . . . .	5
4.3	Entwurf der Datenhaltung . . . . .	5
<b>5</b>	<b>Implementierung</b>	<b>6</b>
5.1	Implementierung der Angular-Komponenten . . . . .	6
5.2	Implementierung der Drag-and-Drop-Funktionalität . . . . .	6
5.3	Integration in das bestehende System . . . . .	6
5.4	Implementierung der Konfiguration . . . . .	6
<b>6</b>	<b>Abnahme</b>	<b>7</b>
6.1	Durchführung der Tests . . . . .	7
6.2	Code Review . . . . .	7
6.3	Abnahme durch den Projektverantwortlichen . . . . .	7
<b>7</b>	<b>Interne Dokumentationen</b>	<b>7</b>
7.1	Erstellung der Entwicklerdokumentation . . . . .	7
7.2	Erstellung des Benutzerhandbuchs . . . . .	7
<b>8</b>	<b>Fazit</b>	<b>8</b>
8.1	Soll-/Ist-Vergleich . . . . .	8
8.2	Ausblick . . . . .	8
8.3	Gelerntes . . . . .	8
<b>A</b>	<b>Anhang</b>	<b>9</b>
A.1	A1 Detaillierte Zeitplanung . . . . .	9

# 1 Einleitung

## 1.1 Projektumfeld

Diese Projektdokumentation beschreibt den Ablauf meines IHK-Projekts im Rahmen meiner Ausbildung zum Fachinformatiker für Anwendungsentwicklung bei der Green Solutions Software GmbH (GSS-GmbH).

Die GSS-GmbH ist ein innovatives Softwareunternehmen mit 50 Mitarbeitern, das Software primär für die grüne Branche anbietet. Die zentrale Software der GSS-GmbH ist die green solutions cloud software (GSCS), welche relevante Kanäle im Online- und Offline-Bereich in einem System zusammenfügt. Von Webshop, Website, Newsletter, Dropshipping, CRM, SEO, SEA und Social Media bis hin zu Beschilderungen, Apps und digitaler Kundenbindung integriert die GSCS diverse Funktionalitäten.

Im Zuge der Modernisierung wurde das Frontend der GSCS von einer MVC-Architektur mit jQuery auf Angular umgestellt. Dadurch ist die bisherige Formularverwaltung nicht mehr kompatibel mit der neuen Architektur. Formulare sind ein zentrales Element der GSCS und werden in verschiedenen Bereichen eingesetzt, von Kundendaten bis hin zu Produktkonfigurationen.

## 1.2 Projektziel

Ziel des Projekts ist es, einen dynamischen Formular-Editor für die GSCS zu entwickeln und zu implementieren. Der Editor soll es Benutzern mit speziellen Rechten (Designer-Rechten) ermöglichen, Formulare innerhalb des Systems flexibel zu gestalten und anzupassen. Die Umsetzung erfolgt mit dem .NET Framework im Backend und Angular im Frontend.

Der Editor wird als Floating Window implementiert und soll folgende Kernfunktionalitäten bieten:

- Hinzufügen und Entfernen von Formularfeldern
- Anpassung der Feldbreite und -größe
- Definition von Pflichtfeldern
- Festlegung von Validierungsregeln
- Speichern von Formular-Templates

Die Änderungen sollen in Echtzeit visualisiert und nach dem Speichern sofort im System verfügbar sein.

## 1.3 Projektbegründung

Die Umstellung des Frontends der GSCS auf Angular erfordert eine Neuentwicklung der Formularbearbeitung. Der bisherige Formular-Editor ist nicht mehr kompatibel mit der neuen Angular-Architektur. Formulare sind jedoch ein zentraler Bestandteil der GSCS und werden in verschiedenen Bereichen wie Kundenverwaltung, Produktkonfiguration und Bestellabwicklung eingesetzt. Eine flexible und benutzerfreundliche Möglichkeit zur Anpassung von Formularen ist für die Kunden der GSS-GmbH von großer Bedeutung, da verschiedene Branchen unterschiedliche Anforderungen an Formulare haben. Die Entwicklung eines neuen, auf Angular basierenden Formular-Editors ist daher unerlässlich, um die Funktionalität der GSCS aufrechtzuerhalten und zu verbessern.

## 1.4 Projektschnittstellen

Die GSCS der GSS-GmbH basiert auf C# mit ASP.NET und dem Entity Framework und wird auf einem IIS-Webserver mit einer SQL-Datenbank betrieben. Das Frontend wurde von MVC mit jQuery auf Angular umgestellt.

Die Benutzer des Formular-Editors sind primär Administratoren und Benutzer mit Designer-Rechten, die für die Anpassung und Erstellung von Formularen innerhalb des Systems verantwortlich sind.

Ein Code Review wurde in Vorbereitung der Abnahme mit einem weiteren Fachinformatiker für Anwendungsentwicklung der GSS-GmbH durchgeführt.

Genehmigt wurde das Projekt von der Geschäftsführung der GSS-GmbH. In regelmäßigen Meetings wurde der Abteilungsleiter der Entwicklung über den aktuellen Entwicklungsstand und den Fortschritt des Projekts informiert.

## 1.5 Projektabgrenzung

Der Formular-Editor wird in das bestehende GSCS-System integriert. Die Implementierung umfasst die Angular-Frontend-Umsetzung und die Integration in das bestehende Backend-System. Folgende Einschränkungen wurden berücksichtigt:

- Die Bearbeitung von Formularen ist nur für Benutzer mit Designer-Rechten möglich
- Bestimmte System-Kern-Formulare können nicht bearbeitet werden
- Die Formularänderungen müssen mit der bestehenden Datenstruktur kompatibel sein

## 2 Planung

### 2.1 Projektphasen

Für das vorliegende Projekt standen insgesamt 80 Stunden für die Umsetzung zur Verfügung. Im Rahmen des Projektantrags wurde bereits eine strukturierte Aufteilung des Projekts in verschiedene Phasen durchgeführt:

1. Planung/Analyse (10h)
2. Entwurf (14h)
3. Implementierung (40h)
4. Testen und Abnahme (8h)
5. Dokumentation (8h)

Die detaillierte Zeitplanung befindet sich im Anhang A1 Detaillierte Zeitplanung.

Die Umsetzung des Projekts erfolgte während der regulären Arbeitszeit. Die Durchführung des Projekts verlief parallel zum Berufsalltag.

## 2.2 Ressourcenplanung

Im Rahmen der Ressourcenplanung wurden für das Projekt alle notwendigen Faktoren wie Personal, Hard- und Software berücksichtigt. Bei der Wahl der eingesetzten Software wurde darauf geachtet, dass diese bereits Bestandteil der erworbenen Lizenzen der GSS-GmbH oder kostenlos erhältlich waren, um zusätzliche Kosten so gering wie möglich zu halten.

Die Entwicklung fand auf zwei PCs statt, einem Office-Rechner mit Windows 11 und 6 Bildschirmen und einem Homeoffice-Rechner mit Windows 11 und 3 Bildschirmen. Bei beiden wurde die Entwicklungsumgebung Visual Studio Professional 2022 verwendet. Für die Angular-Entwicklung kamen außerdem VS Code, Node.js und npm zum Einsatz. Zur Versionsverwaltung des Projekts wurden die bestehenden Git-Repositories der GSS-GmbH verwendet.

## 2.3 Entwicklungsprozess

Die Umsetzung des Projekts verlief nach dem Wasserfallmodell. Die einzelnen Phasen wurden nacheinander abgearbeitet, wobei es in einigen Phasen zu Überlappungen kam, um den Entwicklungsprozess zu optimieren.

# 3 Analyse

## 3.1 Durchführung Ist-Analyse

Das bisherige System nutzte Dotnet als MVC-Architektur, wobei die Formularbearbeitung mit Hilfe von jQuery umgesetzt wurde. Die Formulare wurden in der Datenbank als JSON-Struktur gespeichert und bei Bedarf dynamisch gerendert.

Bestehende Probleme im alten System:

- Keine Vorschau-Funktionalität während der Bearbeitung
- Fehlende Drag-and-Drop-Funktionalität für einfachere Bedienung
- Keine Möglichkeit zur Speicherung von Formular-Templates

## 3.2 Durchführung Soll-Analyse

Das Ziel des Projekts ist es, einen modernen, auf Angular basierenden Formular-Editor zu entwickeln, der die bestehenden Probleme behebt und neue Funktionalitäten bietet.

Folgende Anforderungen wurden definiert:

- Entwicklung eines modernen, benutzerfreundlichen Editors
- Implementierung einer Echtzeit-Vorschau
- Integration in das bestehende Angular-Frontend
- Kompatibilität mit der bestehenden Datenstruktur
- Drag-and-Drop-Funktionalität für einfache Bedienung
- Speichermöglichkeit für Formular-Templates

### 3.3 Durchführung der Wirtschaftlichkeitsanalyse und Amortisationsrechnung

Durch die Modernisierung des Formular-Editors wird die Benutzerfreundlichkeit der GSCS erheblich verbessert. Dies führt zu einer höheren Kundenzufriedenheit.

Da der Formular-Editor ein integraler Bestandteil der GSCS ist und die Umstellung auf Angular bereits beschlossen war, war eine Neuentwicklung unumgänglich. Alternativen wie die Beibehaltung des alten Systems oder der Zukauf einer externen Lösung wurden aufgrund von Kompatibilitätsproblemen und hohen Lizenzkosten verworfen.

Die Kosten für die Entwicklung des Formular-Editors belaufen sich auf ca. 4.000€ (80 Stunden à 50€). Die erwartete Amortisationsdauer beträgt etwa 6 Monate durch gesteigerte Effizienz bei der Formularbearbeitung und reduzierte Supportanfragen.

## 4 Entwurf

### 4.1 Entwurf der Benutzeroberfläche

Die Benutzeroberfläche des Formular-Editors wurde als Floating Window konzipiert, das über dem eigentlichen Formular angezeigt wird. Das Design folgt dem Corporate Design der GSCS und integriert sich nahtlos in die bestehende Benutzeroberfläche.

Die Benutzeroberfläche wurde in folgende Bereiche unterteilt:

- Toolbox mit verfügbaren Formularfeldern (Textfeld, Zahlenfeld, Dropdown, Checkbox, Radio-Button, etc.)
- Konfigurationsbereich für das ausgewählte Formularfeld
- Vorschaubereich für das gesamte Formular
- Aktionsbereich mit Buttons zum Speichern, Abbrechen und Zurücksetzen

### 4.2 Entwurf der Komponentenstruktur

Für die Implementierung des Formular-Editors wurde eine modulare Komponentenstruktur gewählt, die sich gut in die bestehende Angular-Architektur integriert. Folgende Hauptkomponenten wurden definiert:

- FormEditorComponent: Hauptkomponente, die das Floating Window und die Integration der anderen Komponenten handhabt
- ToolboxComponent: Verwaltet die verfügbaren Formularfelder
- FieldConfigComponent: Stellt die Konfigurationsmöglichkeiten für das ausgewählte Formularfeld bereit
- FormPreviewComponent: Zeigt eine Echtzeit-Vorschau des Formulars an

### 4.3 Entwurf der Datenhaltung

Die Formulardaten werden in einer JSON-Struktur gespeichert, die sowohl die Feldtypen als auch deren Konfiguration und Validierungsregeln enthält. Diese Struktur ist kompatibel mit dem bestehenden Datenmodell und kann problemlos in der SQL-Datenbank der GSCS gespeichert werden.

```
{
  "formId": "customer-form",
  "formName": "Kundenformular",
  "fields": [
    {
      "id": "name",
      "type": "text",
      "label": "Name",
      "required": true,
      "width": 12
    },
    {
      "id": "email",
      "type": "email",
      "label": "E-Mail",
      "required": true,
      "width": 6
    }
  ]
}
```

## 5 Implementierung

### 5.1 Implementierung der Angular-Komponenten

Die Implementierung der Angular-Komponenten erfolgte gemäß dem Entwurf. Besonderer Wert wurde auf Wiederverwendbarkeit und Testbarkeit gelegt. Die Komponenten wurden als eigenständige Module implementiert, die unabhängig voneinander getestet werden können.

Die FormEditorComponent wurde als Modal-Dialog implementiert, der über dem eigentlichen Formular angezeigt wird. Die Kommunikation zwischen den Komponenten erfolgt über Services und EventEmitter.

### 5.2 Implementierung der Drag-and-Drop-Funktionalität

Für die Drag-and-Drop-Funktionalität wurde die Angular CDK (Component Development Kit) verwendet. Diese ermöglicht es, Formularfelder einfach per Drag-and-Drop aus der Toolbox in das Formular zu ziehen und innerhalb des Formulars zu verschieben.

### 5.3 Integration in das bestehende System

Die Integration in das bestehende System erfolgte über einen eigenen API-Endpunkt, der die Speicherung und den Abruf von Formulardaten ermöglicht. Die Berechtigungsprüfung erfolgt über das bestehende Rechtesystem der GSCS.

### 5.4 Implementierung der Konfiguration

Die Konfiguration des Formular-Editors erfolgt über eine eigene Konfigurationskomponente, die es ermöglicht, globale Einstellungen wie verfügbare Feldtypen, Standard-Validierungsregeln, etc. festzulegen.

## 6 Abnahme

### 6.1 Durchführung der Tests

Zur Sicherstellung der Qualität wurden umfangreiche Tests durchgeführt:

- Unit-Tests für die einzelnen Komponenten und Services
- Integrationstests für das Zusammenspiel der Komponenten
- End-to-End-Tests für die Gesamtfunktionalität des Formular-Editors

Dabei wurden verschiedene Testszenarien durchgespielt, wie das Anlegen neuer Formulare, die Bearbeitung bestehender Formulare, die Validierung von Eingaben, etc.

### 6.2 Code Review

Vor der Abnahme wurde ein Code Review mit einem weiteren Entwickler der GSS-GmbH durchgeführt. Dabei wurden einige Verbesserungsvorschläge gemacht, die vor der finalen Abnahme umgesetzt wurden.

### 6.3 Abnahme durch den Projektverantwortlichen

Die Abnahme des Projekts erfolgte durch den Projektverantwortlichen Herrn Alexander Kelm. Dabei wurden alle Anforderungen geprüft und für erfüllt befunden. Der Formular-Editor wurde anschließend in die Produktivumgebung übernommen.

## 7 Interne Dokumentationen

### 7.1 Erstellung der Entwicklerdokumentation

Die Entwicklerdokumentation umfasst eine detaillierte Beschreibung der Architektur, der Komponenten und der Schnittstellen des Formular-Editors. Sie dient als Referenz für zukünftige Weiterentwicklungen und Wartungsarbeiten.

Die Dokumentation wurde in Markdown erstellt und im Git-Repository der GSS-GmbH abgelegt. Zusätzlich wurden die Komponenten und Services im Code ausführlich dokumentiert.

### 7.2 Erstellung des Benutzerhandbuchs

Das Benutzerhandbuch beschreibt die Bedienung des Formular-Editors aus Sicht der Endbenutzer. Es enthält detaillierte Anleitungen mit Screenshots zur Erstellung und Bearbeitung von Formularen, zur Konfiguration von Formularfeldern und zur Festlegung von Validierungsregeln.

Das Benutzerhandbuch wurde im Wiki-System der GSS-GmbH veröffentlicht und steht allen Mitarbeitern und Kunden zur Verfügung.



## 8 Fazit

### 8.1 Soll-/Ist-Vergleich

Die Umsetzung des Formular-Editors wurde erfolgreich abgeschlossen und alle definierten Anforderungen wurden erfüllt. Die Zeitplanung wurde eingehalten, lediglich in der Implementierungsphase gab es leichte Verzögerungen durch unvorhergesehene Schwierigkeiten bei der Integration in das bestehende System, die jedoch durch effizientere Arbeit in den anderen Phasen ausgeglichen werden konnten.

In der Testphase wurden einige kleinere Fehler entdeckt, die vor der finalen Abnahme behoben wurden. Insgesamt verlief das Projekt nach Plan und die Qualität des Endergebnisses entspricht den Erwartungen.

### 8.2 Ausblick

Der Formular-Editor wird in zukünftigen Versionen der GSCS weiterentwickelt. Geplante Erweiterungen umfassen:

- Erweiterung um zusätzliche Feldtypen (Datei-Upload, Datum/Uhrzeit, etc.)
- Verbesserung der Vorschau-Funktionalität
- Integration von Formular-Templates für verschiedene Branchen
- Implementierung einer Versionierung für Formulare

### 8.3 Gelerntes

Während der Projektumsetzung konnte ich viele Erfahrungen im Bereich der Angular-Entwicklung sammeln. Die Arbeit mit Angular und TypeScript hat mir gezeigt, wie moderne Web-Frameworks die Entwicklung komplexer Anwendungen erleichtern können.

Die Herausforderungen bei der Integration in das bestehende System haben mein Verständnis für die Komplexität großer Softwareprojekte vertieft. Die enge Zusammenarbeit mit dem Projektverantwortlichen und anderen Entwicklern hat mir wertvolle Einblicke in die Projektorganisation und -kommunikation gegeben.

## A Anhang

### A.1 A1 Detaillierte Zeitplanung

Phase	Aktivität	Zeit (h)
Planung/Analyse	Durchführung Ist-Analyse	2
	Durchführung Soll-Analyse	5
	Analyse bestehender Formular Strukturen	1
	Erstellung Lastenheft	2
Entwurf	Entwurf der Benutzeroberfläche	6
	Planung der Komponentenstruktur	4
	Konzeption der State Management Strategie	2
	Erstellung Pflichtenheft	2
Implementierung	Modaler Editor	12
	Formular-Konfiguration	12
	System-Integration	8
	Validierungslogik	8
Testen und Abnahme	Entwicklung Testszenarien	2
	Durchführung Tests	2
	Code-Review	2
	Abnahme	2
Dokumentation	Technische Dokumentation	4
	Benutzerhandbuch	4
<b>Gesamt</b>		<b>80</b>