

# Technische Dokumentation



<b>Challengegeber</b>	Bracher und Partner
<b>Challenge Teilnehmer</b>	Mete Turna Pascal Knecht Aaron Giaimo Finn Fiedler
<b>Projekt Start / End</b>	23.08.24 / 25.08.24
<b>Projektname</b>	EasyHreg
<b>Status</b>	Freigegeben

## Bearbeitungshistorie:

<b>Name</b>	<b>Datum</b>	<b>Beschreibung</b>
Mete Turna	24.08.24	Ersterfassung

# Inhaltsverzeichnis

<b>Angewandte Technologie</b>	<b>3</b>
<b>Hosting</b>	<b>3</b>
<b>Entwicklung</b>	<b>3</b>
Backend	3
Frontend	3
<b>Ausgangslage</b>	<b>3</b>
<b>Technischer Aufbau</b>	<b>4</b>
<b>Implementation</b>	<b>4</b>
<b>Abgrenzung</b>	<b>4</b>

# Angewandte Technologie

## Programmiersprachen

Das Projekt wurde in C# (Backend), JavaScript (Frontend) programmiert.

## Hosting

Das Backend hosten wir auf Azure.

Hinweis: Der Host wird am 26.08.24 ausgeschaltet.

## Entwicklung

### Backend

Repository: <https://github.com/int0x81/EasyHreg.Backend.git>

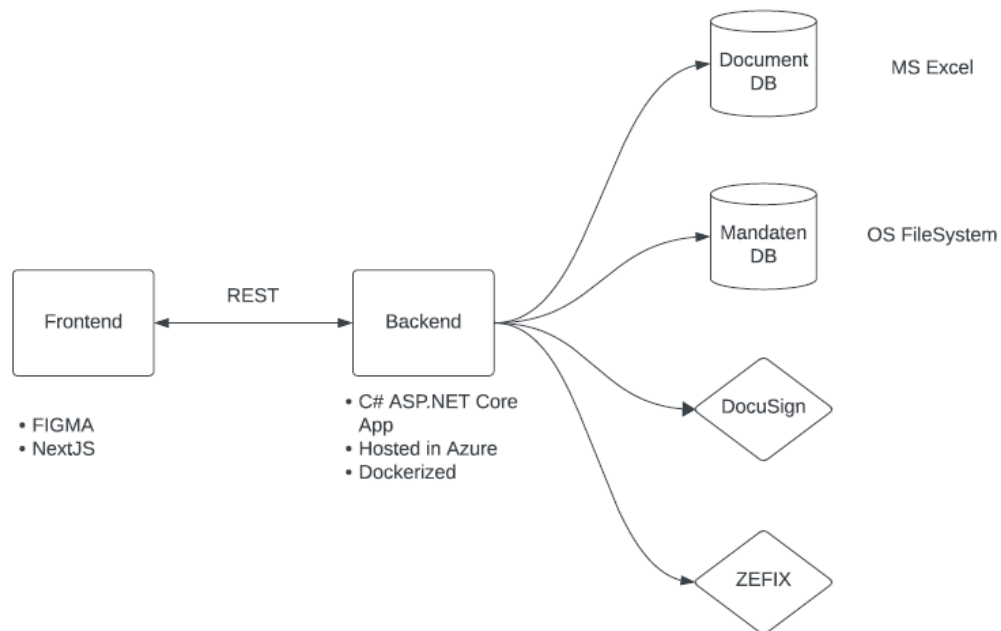
### Frontend

Repository: <https://github.com/Dantolos/BernHackt-Frontend.git>

## Ausgangslage

Zurzeit ist viel administrativer Aufwand nötig um z.B. eine Mutation im Handelsregister vorzunehmen. Einzelnen Unterschriften musste hinterher gerannt werden und ein unnötiger grosser Austausch (Ping-Pong) zwischen der eingetragenen Firma und der Anwaltskanzlei/Notar ist nötig, bis alle Dokumente korrekt eingereicht sind. Genau diesen unnötigen, grossen administrativen Overhead soll unsere Software angehen und lösen.

## Technischer Aufbau



## Implementation

- Wir konnten innerst kürzester Zeit ein funktionierendes Produkt nach Best Practices implementieren, welches in der Cloud gehostet und durch sein intuitives UI überzeugt

## Abgrenzung

- Zefix: Gerne hätten wir öffentlich zugängliche Daten aus dem Handelsregister (Zefix) direkt über eine API abgegriffen. Das Zefix bietet dazu sogar eine Rest-API an, allerdings braucht man dazu Test-Login-Credentials, welche wir leider am Wochenende nicht erlangen konnten. Deshalb haben wir diese Daten/Service gemockt.
- DocuSign: Auch wollten wir die Möglichkeit bieten, dass unsere Lösung eine digitale Signatur verifizieren kann. In unserer Recherche sind wir auf DocuSign gestoßen, welche versprechen mit ihren Dev-Tools eine solche Möglichkeit gleich in die Applikation zu implementieren. In der Implementierungsphase haben sich leider grössere technische Schwierigkeiten ergeben, als vorab ersichtlich waren: z.B. erfordern Callbacks von der Plattform https-Domains, was unser Service im Rahmen dieses POC noch nicht anbietet.