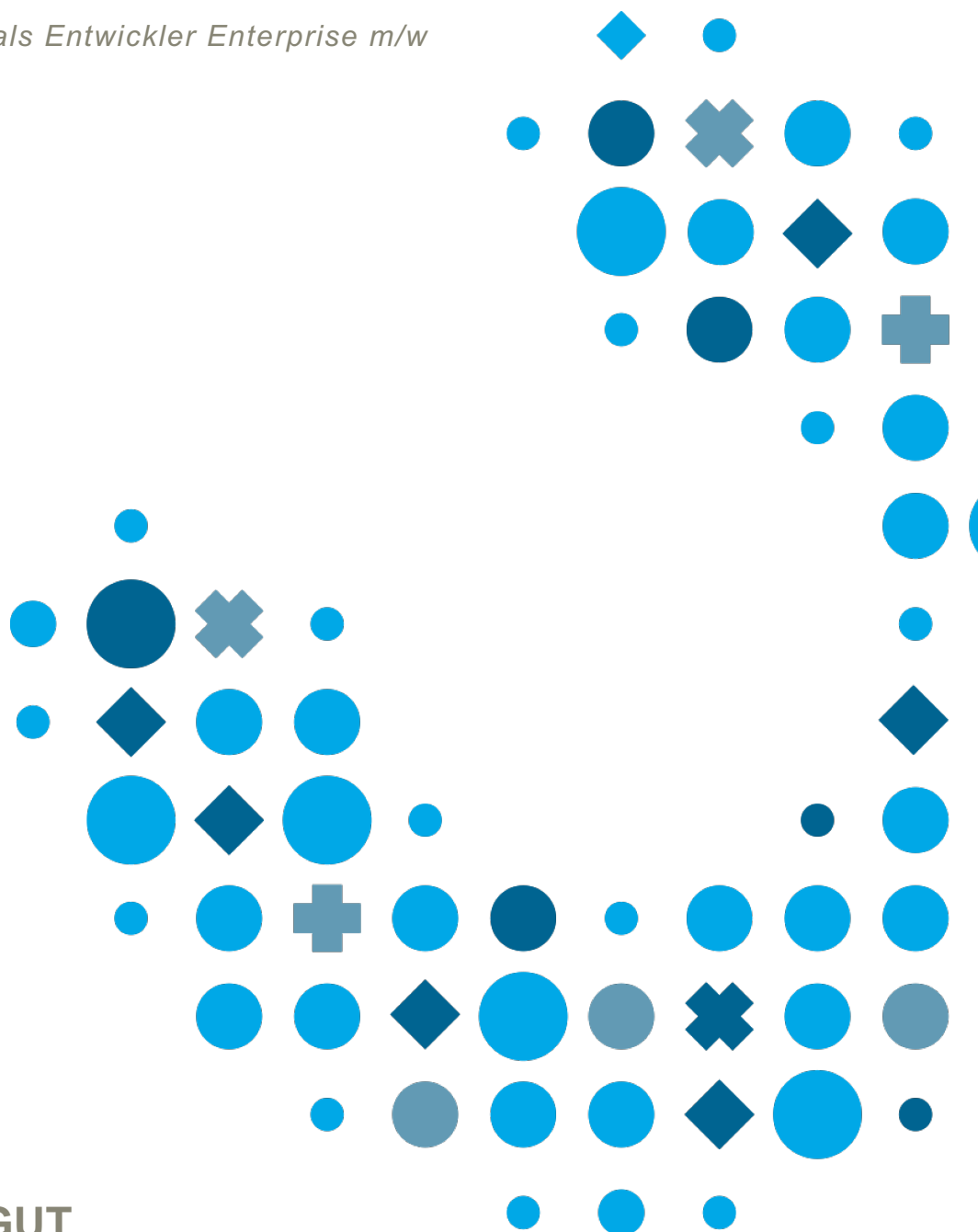


ToDo-Liste

Aufgabe für Bewerber als Entwickler Enterprise m/w

Stand: Oktober 2018



Inhaltsverzeichnis

1	Aufgabenbeschreibung.....	3
2	User Stories	4
3	Kontakt	12

1 Aufgabenbeschreibung

offenkundig ist nicht nur unser Motto, sondern auch unser Anspruch an uns selbst. Wir möchten sehen, dass auch du offen, kundig und gut bist und mit dieser Aufgabe hast du dazu die Gelegenheit.

Wir möchten, dass du eine kleine Anwendung - eine Todo-Liste - programmierst und uns deine Lösung in deinem Bewerbungsgespräch vorstellst. Die Todo-Liste besteht aus einem Service, der über eine Schnittstelle Daten bereitstellt und einer App, die mit dem Service kommuniziert.

Für die Stelle als Entwickler Enterprise besteht deine Aufgabe in der Implementierung des Service in Java oder JavaScript. Um dem ganzen einen Rahmen zu geben, haben wir die Anforderungen an die Todo-Liste in Form einiger User Stories beschrieben. Jede User Story enthält eine Reihe von Akzeptanzkriterien, die als Szenarien formuliert sind. Zusätzlich stellen wir dir eine dokumentierte Schnittstellendefinition (OpenAPI v2 aka Swagger) bereit.

Für die Implementierung möchten wir dir keine Vorgaben machen, sondern sind gespannt auf deine Lösung. Diese sollte deinem aktuellen Wissenstand und deiner Arbeitsweise entsprechen.

Hinweis: Die User Stories sind nicht priorisiert. Es liegt bei dir eine geeignete Reihenfolge festzulegen. Solltest du feststellen, dass du nicht alle User Stories bis zum Bewerbungsgespräch abschließen kannst, priorisiere diese bitte frühzeitig. Eine lauffähige 80% Lösung ist uns lieber als 100% die nicht funktionieren.

In unserem Arbeitsalltag ist die Kommunikation mit unterschiedlichen Ansprechpartnern beim Kunden (Managern, Fachexperten, Entwicklern, ...) von besonderer Bedeutung. Das Bewerbungsgespräch bietet dir daher den idealen Rahmen, um deine Lösung zu zeigen. Für die Präsentation stehen dir bis zu 15 Minuten zur Verfügung.

Bevor du uns in einer kurzen Live-Demo deine Implementierung demonstrierst und anhand einiger ausgewählter Code-Stellen deinen Code-Style vorstellst, möchten wir, dass du uns erklärst, welche fachlichen Features deine Todo-Liste uns bietet und warum wir diese nutzen sollten. Hierfür kannst du auch gerne ein paar Folien vorbereiten. Damit wir deine Lösung schon mal ansehen können, stell uns bitte deinen Quellcode im Vorfeld deines Gesprächs als Zip-Datei zur Verfügung.

Hinweis: Sofern du kein Notebook für die Präsentation besitzt, informiere uns bitte rechtzeitig, damit wir alles Nötige abstimmen und vorbereiten können.

2 User Stories

#1 Ein Todo erstellen

Als Nutzer

möchte ich ein Todo erstellen

um mich an eine zu erledigende Aufgabe zu erinnern

Akzeptanzkriterien:

- Wenn ein Todo mit vollständigen, gültigen Daten übermittelt wird, wird es angelegt
- Wenn ein Todo mit unvollständigen Daten übermittelt wird, wird es nicht angelegt
- Wenn ein Todo mit ungültigen Daten übermittelt wird, wird es nicht angelegt

Scenario: Requesting api operation to create a new todo

Given a todo with the following properties

```
| title | Clean fridge |
| description | It's a mess |
| dueDate | date |
| done | false |
```

When a client wants to create a new todo

Then the status code is 201

And response includes the following in any order

```
| id | 1 |
| title | Clean fridge |
| description | It's a mess |
| dueDate | date |
| done | false |
```

Scenario Outline: Requesting api operation to create a new todo with incomplete data

Given a todo without <param>

When a client wants to create a new todo

Then the status code is 400

And error response includes the following in any order

```
| code | <code> |
| message | <message> |
```

Examples:

```
| param | code | message |
| title | TITLE_NULL | title must not be null |
| dueDate | DUE_DATE_NULL | dueDate must not be null |
```

Scenario Outline: Validation Error Responses

When a client wants to create a todo with <param>

Then the status code is 400

And error response includes the following in any order

```
| code      | <code> |
| message   | <message> |
```

Examples:

```
| param      | code      | message
| title=""   | TITLE_SIZE | title size must be between 1 and 30 |
| title="Lorem ipsum dolor sit amet, con" | TITLE_SIZE | title size
must be between 1 and 30 |
| description="Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis
natoque penatibus et magnis dis parturient montes, nascetur ridiculus
mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis,
sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel,
aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet
a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.
Integer tincidunt. Cras dapibus" | DESCRIPTION_SIZE | description size
must be between 0 and 500 |
| dueDate="18-12-12T12:12:12Z" | DUE_DATE_PATTERN | dueDate pattern
must match "yyyy-MM-ddTHH:mm:ssZ" |
```

#2 Ein Todo ändern

Als Nutzer

möchte ich ein bestehendes Todo ändern

um die Beschreibung einer zu erledigenden Aufgabe anzupassen

Akzeptanzkriterien:

- Wenn das Todo existiert und die neuen Todo-Daten vollständig und gültig sind, wird das Todo geändert
- Falls kein Todo mit der angefragten ID existiert, gibt es einen Fehler
- Wenn die neuen Todo-Daten unvollständig sind, wird das Todo nicht geändert
- Wenn die neuen Todo-Daten ungültig sind, wird das Todo nicht geändert

Background

Given a todo with the following properties

```
| id | 1 |
| title | clean fridge |
| description | It's a mess |
| dueDate | date-1 |
| done | false |
```

Scenario: Requesting api operation to update an existing todo

When a client wants to mark the todo with id 1 as done

And change the title to "clean kitchen"

Then the status code is 204

And the todo should look like this

```
| id | 1 |
| title | clean kitchen |
| description | It's a mess |
| dueDate | date-1 |
| done | true |
```

Scenario: Todo doesn't exist

When a client wants to change the todo with id -1

Then the status code is 404

Scenario Outline: Requesting api operation to update a todo with incomplete data

When a client wants to update the todo with id 1

And the update doesn't contain a <param>

Then the status code is 400

And error response includes the following in any order

```
| code | <code> |
| message | <message> |
```

And the todo should not be updated

Examples:

```
| param | code | message |
| title | TITLE_NULL | title must not be null |
| dueDate | DUE_DATE_NULL | dueDate must not be null |
```

Scenario Outline: Validation Error Responses

When a client wants to update the todo with id 1

And the update contains <param>

Then the status code is 400

And error response includes the following in any order

```
| code | <code> |
| message | <message> |
```

And the todo should not be updated

Examples:

```
| param | code | message |
| title="" | TITLE_SIZE | title size must be between 1 and 30 |
| title="Lorem ipsum dolor sit amet, con" | TITLE_SIZE | title size must be between 1 and 30 |
| description="Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel,
```

```
aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet
a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.
Integer tincidunt. Cras dapibus" | DESCRIPTION_SIZE | description size
must be between 0 and 500 |
    | dueDate="18-12-12T12:12:12Z" | DUE_DATE_PATTERN | dueDate pattern
must match "yyyy-MM-ddTHH:mm:ssZ" |
```

#3 Ein Todo löschen

Als Nutzer

möchte ich ein Todo löschen

um mich nicht mehr an eine zu erledigende Aufgabe zu erinnern

Akzeptanzkriterien:

- Wenn ein Todo mit der übergebenen ID existiert, wird das Todo gelöscht
- Wenn kein Todo mit der übergebenen ID existiert, wird ein Fehler zurückgegeben

Scenario: Requesting api operation to delete an existing todo

Given an existing todo

When a client wants to delete the todo with id 1

Then the status code is 204

Scenario: Requesting api operation to delete a non-existing todo

Given an existing todo

When a client wants to delete the todo with id -1

Then the status code is 404

#4 Ein Todo abfragen

Als Nutzer

möchte ich ein Todo abfragen

um mir die Details einer zu erledigenden Aufgabe durchzulesen

Akzeptanzkriterien:

- Wenn ein Todo mit der gewünschten ID existiert, wird dieses zurückgegeben
- Wenn kein Todo mit der gewünschten ID existiert, wird Status 404 zurückgegeben

Scenario: Requesting api operation to get an existing todo

Given the following todo

id	1
title	clean fridge
description	It's a mess
dueDate	date-1
done	false

When a client wants to retrieve the todo with id 1

Then the status code is 200

And response includes the following in any order

id	1
title	clean fridge
description	It's a mess
dueDate	date-1
done	false

Scenario: Todo doesn't exist

When a client wants to retrieve the todo with id -1

Then the status code is 404

#5 Eine Liste von Todos abfragen

Als Nutzer

möchte ich eine Liste von Todos abfragen

um mir einen Überblick über meine unerledigten Aufgaben zu verschaffen

Akzeptanzkriterien:

- Wenn bereits Todos existieren, werden diese als Liste zurückgegeben
- Wenn noch keine Todos existieren, wird Statuscode 204 zurückgegeben

Scenario: Requesting api operation to get a list of all todos

Given these todos:

id	title	description	dueDate	done
1	clean fridge	It's a mess	date-1	false

	2		clean bathroom				date		false	
	3		bring out garbage		plastic, paper		date+1		false	
	4		go to supermarket		buy milk & eggs		date+3		true	
	5		write user stories		including scenarios		date+5		false	
	6		pay bills				date+5		false	
	7		call mum				date+5		false	

When a client wants to retrieve the todo list

Then the status code is 200

And response includes the following in any order

	id		title		dueDate		done			
	1		clean fridge		date-1		false			
	2		clean bathroom		date		false			
	3		bring out garbage		date+1		false			
	4		go to supermarket		date+3		true			
	5		write user stories		date+5		false			
	6		pay bills		date+5		false			
	7		call mum		date+5		false			

Scenario: Requesting api operation to get an empty list of todos

Given an empty list of todos

When a client wants to retrieve the todo list

Then the status code is 204

#6 Eine Liste von gefilterten Todos abfragen

Als Nutzer

möchte ich eine Liste von gefilterten Todos abfragen
um die für mich relevantesten Aufgaben anzusehen

Akzeptanzkriterien:

- Mit der Filterauswahl state="unfinished" (default) wird eine Liste mit nicht-erledigten Todos zurückgegeben
- Mit der Filterauswahl state="all" wird eine Liste mit erledigten und nicht-erledigten Todos zurückgegeben
- Mit dem Parameter limit=n (default n = 5) wird eine Liste mit maximal n Todos zurückgegeben
- Mit dem Parameter offset=m (default m = 0) wird eine Liste von Todos ab dem m+1-ten Element zurückgegeben
- Ist die Liste der Todos leer wird der Statuscode 204 zurückgegeben
- Existieren weitere noch nicht geladene Todos wird der Statuscode 206 zurückgegeben
- Existieren keine weitere noch nicht geladene Todos wird der Statuscode 200 zurückgegeben

Background:

Given these todos:

id	title	description	dueDate	done
1	clean fridge	It's a mess	date-1	false
2	clean bathroom		date	false
3	bring out garbage	plastic, paper	date+1	false
4	go to supermarket	buy milk & eggs	date+3	true
5	write user stories	including scenarios	date+5	false
6	pay bills		date+5	false
7	call mum		date+5	false

Scenario: Requesting api operation to get a list of todos with default parameters

When a client wants to retrieve todos without given state param

And limit isn't given

And offset isn't given

Then the status code is 206

And response includes the following in any order

1	clean fridge	date-1	false
2	clean bathroom	date	false
3	bring out garbage	date+1	false
5	write user stories	date+5	false
6	pay bills	date+5	false

Scenario: Get a list of all todos with parameters

When a client wants to retrieve todos with state="all"

And limit is 5

```

And offset is 3
Then the status code is 200
And response includes the following in any order
  | 4 | go to supermarket | date+3 | true |
  | 5 | write user stories | date+5 | false |
  | 6 | pay bills          | date+5 | false |
  | 7 | call mum           | date+5 | false |

```

Scenario: Get an empty list of todos

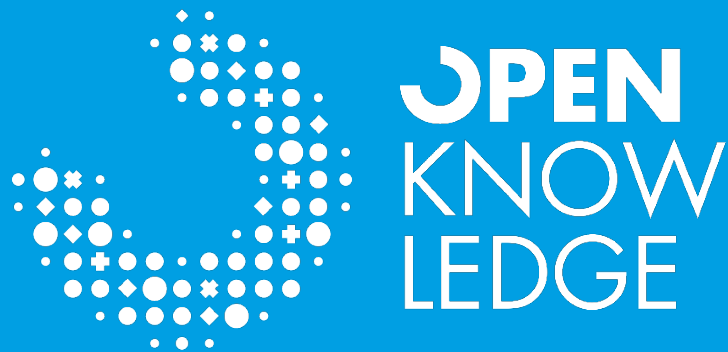
```

When a client wants to retrieve todos with state="unfinished"
And limit is 1
And offset is 10
Then the status code is 204

```

(Diese Story erweitert Story #5)

3 Kontakt



Kontakt

OPEN KNOWLEDGE GmbH
Poststraße 1, 26122 Oldenburg
Tel.: +49 441 – 4082-0
Fax: +49 441 – 4082-111
kontakt@openknowledge.de