Modul 431 IPERKA

# Idee

Ich habe mir überlegt das ich eine kleine Todo Applikation machen will. Das Ziel ist, das man ein Todo erstellen, lesen, bearbeiten und löschen kann (CRUD). Was ich auch noch machen will ist das das Todo in einer Datenbank gespeichert wird. Auf der Index Seite werden alle Todos angezeigt es gibt verschieden Routen eine, um das Todo zu bearbeiten, eine, um das Todo zu löschen und eine, um ein Todo zu erstellen. Das Ganze werde ich in Java Spring Boot umsetzten. Ich mache das Projekt allein also ohne Partner.

Inhalt

[Idee 1](#_Toc103870607)

[Planung 2](#_Toc103870608)

[Create Todo 2](#_Toc103870609)

[Reade Todo 2](#_Toc103870610)

[Update Todo 2](#_Toc103870611)

[Delete Todo 2](#_Toc103870612)

[Projekt Dokumentation 2](#_Toc103870613)

# Planung

Ich werde die Planung an meinen User Stories orientiert ich habe die User Stories in vier Kategorien aufgeteilt create, read, update und delete.

Als nächste mach ich mir ein Neus Spring Boot Projekt inzwischen weiss ich auch schon, wie das geht und welche Dependencies ich brauche weiss ich auch.

### Datenbank

* Wie man eine Datenbank einbindet, muss ich mich noch informieren.
* Die Datenbank braucht aber nur eine Tabelle, weil ich nur Todos speichern will.

### Create Todo

* Eine Route, um ein Todo zu erstellen.
* Eine View, auf der man ein Todo erstellen kann, die /createtodo Route führt auf diese View.
* Wenn man auf der /createtodo Route ist und ein neues Todo erstellt muss dieses Todo in der Datenbank gespeichert werden.
* Wenn man auf der Index Seite ist, kann man auf einen Link klicken der zu der /createtodo Route führt.

### Reade Todo

* Auf der Index Seite eine View die alle Todos anzeigt.
  + Ich bin mir noch nicht sicher, wie ich das machen werde. Ich weiss ich muss mir alle Todos aus der Datenbank holen und dann in einem Modell speichern und mit Thymeleaf und einer each Schleife durch Iterieren und in einem Form anzeigen lassen. Ich bin mir aber nicht sicher ob das so funktioniert ich muss mir nochmals Gedanken dazu machen.
* Eine zweite Route, um die Todos zu sehen die /Todos heisst.
  + Ich denke ich kann, wenn ich die Viewfuction für die / route schreibe auf der alle Todos angezeigt werden auch einfach die /Todos Route hinschreiben damit ich die gleiche Viewfuction verwenden kann.

### Update Todo

* Eine Route, um das Todo zu bearbeiten.
* Auf der Index Seite gibt es einen Link der zu der /edittodo Route führt.
* Das Editierte Todo muss in der Datenbank gespeichert werden.

### Delete Todo

* Eine Route, um Todos zu löschen.
  + Wenn man ein Todo löscht, muss es aus der Datenbank gelöscht werden.
* Auf der Index Seite gibt es einen Link, um ein Todo zu löschen.

# Projekt Dokumentation