



FWE | Hausaufgaben | SS 2024

Bewertung - Beide Hausaufgaben tragen zu jeweils 15% der Gesamtnote bei. Insgesamt können pro Hausaufgabe 100 Punkte erreicht werden. Eine Bewertung mit weniger als 50 Punkten in einer Hausaufgabe führt zum Ausschluss von der Lehrveranstaltung.

Bearbeitungszeit - Beginn **16.04.2024** - Ende **02.06.2024** (bis 23:59 Uhr)

⚠ Code, der **nach** der Bearbeitungszeit eingereicht wird, wird **ausnahmslos nicht** bewertet.

Modus - Die Hausaufgabe muss **einzel** bearbeitet werden.

Die Aufgabe

Hausaufgabe 1 - Backend

Allgemein

Features & Funktionalitäten

Hausaufgabe 2

Allgemein

Features & Funktionalitäten

Generelle Informationen

Abgabe der Aufgabe

Bewertungskriterien

Hausaufgabe 1 (Gesamt - 100 Punkte)

Hausaufgabe 2 (Gesamt - 100 Punkte)

Die Aufgabe

In einer Welt voller Möglichkeiten und Abenteuer ist die Planung einer Reise oft eine Herausforderung. Von der Auswahl des Reiseziels bis hin zur Organisation von Aktivitäten vor Ort - es gibt viele Details zu bedenken. Unsere Anwendung zielt darauf ab, die Reiseplanung zu erleichtern. Mit einem umfassenden CRUD-System für Reisen und Reiseziele ermöglichen wir es Benutzern, Abenteuer einfach und effizient zu planen und zu organisieren.

Hausaufgabe 1 - Backend

Allgemein

In dieser Hausaufgabe konzentrieren wir uns zunächst ausschließlich auf das Backend unserer App. **Verwende ausschließlich Node in Kombination mit TypeScript.** Als Webserver empfehlen wir Express, aber Du hast die freie Wahl. Nutze alle Bibliotheken, die Dir bei Deiner Aufgabe helfen. Zur Speicherung der Daten kannst Du zwischen verschiedenen Varianten wählen, wie z.B. mysql, postgres, mongodb, redis, json, usw.

Features & Funktionalitäten

Die folgenden Dinge sollen umgesetzt werden

- Implementiere ein **CRUD** System, um über eine HTTP-API **Reisen und Reiseziele** anzuzeigen, anzulegen, zu ändern und zu löschen.
- Eine Reise ist ein Container für die geplanten Ziele während der Reise. Dabei können Felder wie Reisezeitraum, Name, Beschreibung, ein Bild, Teilnehmer der Reise, usw. enthalten sein.
- Ein Reiseziel ist ein Ort, der auf der Reise besucht werden soll. Dabei können Felder wie Name, Beschreibung, Zeitraum, Aktivitäten, Fotos, usw. enthalten sein
- Eine Reise kann 0 oder N Reiseziele haben, und ein Reiseziel kann 1 oder N Reisen haben.
- Implementiere die Möglichkeit, ein oder mehrere Reiseziele einer Reise hinzuzufügen und zu löschen.
- Biete eine Route an, die alle Reisen zurückgibt, die ein bestimmtes Reiseziel enthalten.

- Implementiere eine Suchfunktion, um Reisen nach ihrem Namen oder Datum zu durchsuchen.
- Erstelle eine detaillierte Dokumentation der Anwendung in der README-Datei. Diese sollte mindestens die Funktionalitäten beschreiben, die Struktur der Routen auflisten und beschreiben, wie die Anwendung eingerichtet und getestet werden kann.
- Stelle eine Möglichkeit bereit, das System zu testen. Bevorzugt sind automatisierte Tests, aber auch eine Postman-Collection oder curl-Aufrufe sind akzeptabel. Alle funktionalen Fälle müssen abgedeckt sein.
- Es dürfen beliebig viele Felder und weitere Tabellen hinzugefügt werden, solange sie zur App passen und eine sinnvolle Funktionalität bieten.
- Das System ist "Single User" und muss daher keine Accounts, Sessions, Zugriffsrechte, Logins oder Ähnliches beinhalten.

Hausaufgabe 2

Allgemein

In der zweiten Hausaufgabe geht nun um das Frontend, welche das Backend aus der ersten Hausaufgabe erweitert. Nutze den Code aus Hausaufgabe 1 als Grundlage für Hausaufgabe 2. Zur Abnahme muss sowohl das Backend als auch das Frontend **aus einem Repository** heraus startbar sein. Bitte nutzt ausschließlich React und Typescript.

Features & Funktionalitäten

Die folgenden Dinge sollen umgesetzt werden:

- Erstelle eine Listenansicht und eine Detailansicht für die Reisen. Aus den jeweiligen Ansichten sollen sich die Daten anlegen, ändern und ggf. löschen lassen (**CRUD**).
- Implementiere ein **CRUD** System, um über die REST-API aus der ersten Hausaufgabe Reisen und Reiseziele anzuzeigen, anzulegen, zu ändern und zu löschen.
- Biete eine Möglichkeit, Reisen nach ihrem Namen oder ihrem Datum zu durchsuchen, indem du die entsprechende API-Route integrierst.
- Biete eine Möglichkeit, alle Reisen zu einem Reiseziel anzuzeigen.

- Schreibe eine genaue Dokumentation der gesamten Applikation (Front- als auch Backend) in der README.md, welche mindestens
 - die Funktionalitäten beschreibt
 - die Struktur der Routen auflistet
 - beschreibt wie die Applikation aufzusetzen ist
 - beschreibt wie die API getestet werden kann.
- ⚠ Die Dokumentation **muss** in Markdown verfasst werden

Generelle Informationen

Bitte hinterlasse uns eine **genaue Anleitung** zum Aufsetzen in der ReadME deines Projekts. Gerne kannst Du auch **Docker Container** verwenden und uns das Backend/Frontend in einem fertigen Container zur Verfügung stellen, welchen Du in die Gitlab Registry oder den Docker hub pushst. Bitte denke auch in diesem Fall an eine **genaue Anleitung**.

⚠ **node_modules, environment files und api keys** gehören im Normalfall **nicht** in ein Repository, bitte schaut Euch dazu **.gitignore** files an und nutzt sie entsprechend.

Eine README.md File in einem Projekt ist hauptsächlich für andere Entwickler gedacht, welche Euer Projekt aufsetzen und nutzen möchten und sollte daher die entsprechenden Informationen beinhalten. Einen guten Artikel dazu gibt es hier: <https://goo.gl/ZNh6yh>.

Externe APIs für Euren Freestyle Part können z.B. hier gefunden werden: <https://github.com/toddmotto/publicapis> und sollten passend zu der Hausaufgabe ausgewählt werden. Ihr dürft euch auch gerne eine eigene zweite API schreiben über die Ihr Daten konsumiert.

Abgabe der Aufgabe

Der gesamte Code muss in das Gitlab der h_da gepushed werden:
<https://code.fbi.h-da.de/>.

Wie Du dabei Git verwendest, ist Dir überlassen. Wichtig ist nur, dass Du Deine Ergebnisse vor dem Abgabetermin in den **main** Branch eincheckst. Abgaben als .zip Datei oder per E-Mail sind unzulässig und werden **ausnahmslos nicht**

bewertet. In der Vergangenheit kam es hier des Öfteren zu Problemen. Kümmert Euch bitte rechtzeitig darum.

Der Name der ersten und zweiten Hausaufgabe muss folgendem Schema entsprechen **FWE-SS-24-123456** Bitte ersetzt das 123456 durch Eure Matrikelnummer. Beide Hausaufgaben werden **im gleichen Repository** abgegeben.

WICHTIG: Achtet darauf dass Eure Repository auf PRIVATE gestellt ist und Ihr die folgenden beiden Nutzer eingeladen habt @fbi1483 und @hda10128

Bewertungskriterien

Hausaufgabe 1 (Gesamt - 100 Punkte)

Dokumentation & Funktionalität des Projekts - (20 Punkte)

- Hast Du alle benötigten Informationen über Dein Projekt beschrieben?
- Funktioniert das Projekt nach dem Aufsetzen korrekt?

Einheitliche Syntax & moderner modularer Code - (10 Punkte)

- Wie sind Deine Funktionen/Variablen benannt?
- Wie lang sind Deine Funktionen?
- Wie ist Dein Projekt strukturiert?
- Wie interagieren die Klassen miteinander?
- Genügt die Syntax einem Codingstandard?
- Gleichmäßiges Spacing?
- Kein Mix von Sprachen in Klassen, Variablen und Funktionsnamen (deutsch/englisch)?
- Kommentierter Code an den notwendigen Stellen?

Kernfunktionalität - (25 Punkte)

- Ist das Anlegen, Bearbeiten, Auflisten und Löschen von Reisen möglich?
- Ist das Anlegen, Bearbeiten, Auflisten und Löschen von Reisezielen möglich?
- Ist es möglich Reiseziele zu Reisen hinzuzufügen und sie zu entfernen?
- Können alle Reisen die zu einem Reiseziel gehören angezeigt werden?
- Können Reisen nach Ihrem Namen oder Reisedatum gesucht werden?

Fehlerbehandlung - (10 Punkte)

- Hast Du an alle möglichen Nebeneffekte gedacht?
- Was passiert, wenn nicht alle Felder gesendet werden?
- Was passiert, wenn leere Felder gesendet werden?
- Was passiert, Inkorrekte Felder, Script code oder HTML Code gesendet wird?

Verwendung von HTTP - (5 Punkte)

- Ist Dein API Design nach Rest Best Practices aufgebaut?
- Hast Du korrekte Response Codes?

Tests - (5+5 Punkte bei automatisierten Tests)

- Wie testest Du deine Hausaufgabe?
- Beschreibe das genaue Vorgehen und gib uns die Möglichkeit Dein Projekt ebenfalls zu testen ▲ Nur durch die Nutzung von automatisierten Tests ist hier die volle Punktzahl zu erreichen

▲▲▲ WICHTIG: Beschreibe Freestyle Task#1 & #2 in der readme genau (sonst kann es dafür keine Punkte geben). ▲▲▲

Freestyle task #1 - (10 Punkte)

- Überrasche uns mit einem Feature, welches nicht in der Aufgabenbeschreibung gefordert ist. Wichtig: Es muss zu dem aktuellen Projekt passen. Dieses Feature **darf keine Daten von einer externen API konsumieren** sondern muss die Funktionalität der bestehende App erweitern.

Freestyle task #2 - (10 Punkte)

- Überrasche uns mit einem weiteren Feature, welches nicht in der Aufgabenbeschreibung gefordert ist. Wichtig: Es muss zu dem aktuellen Projekt passen. Dieses Feature **muss Daten von einer externen API konsumieren**.

Hausaufgabe 2 (Gesamt - 100 Punkte)

Dokumentation & Funktionalität des Projekts - (20 Punkte)

- Hast Du alle benötigten Informationen über Dein Projekt beschrieben?
- Funktioniert das Projekt nach dem Aufsetzen korrekt?

Einheitliche Syntax & moderner modularer Code - (10 Punkte)

- Wie sind Deine Funktionen/Variablen benannt?
- Wie lang sind Deine Funktionen?
- Wie ist Dein Projekt strukturiert?
- Wie interagieren die Klassen miteinander?
- Genügt die Syntax einem Codingstandard?
- Gleichmäßiges Spacing?
- Kein Mix von Sprachen (deutsch/englisch)?
- Kommentierter Code an den notwendigen Stellen?

Kernfunktionalität - (30 Punkte)

- Reisen erstellen / bearbeiten / löschen / in Liste darstellen

- Reiseziele zu Rezepten hinzufügen & löschen

Fehlerbehandlung - (10 Punkte)

- Hast Du an alle möglichen Nebeneffekte gedacht?
- Was passiert, wenn nicht alle Felder gesendet werden?
- Leere Felder gesendet werden?
- Inkorrekte Felder, Script code oder HTML Code gesendet wird?

Benutzung von Komponenten, Routern - (10 Punkte)

- verwende `react-router` um zwischen den unterschiedlichen Ansichten zu wechseln
- Listenansicht / Detailansicht von Rezepten

Such Funktionalität - (20 Punkte)

- Biete eine Möglichkeit nach Reisen mittels Name oder Datum zu suchen
- Biete eine Möglichkeit alle Reisen zu einem bestimmten Reiseziel anzuzeigen