

Projektarbeit Modul 294

Entwickeln Sie mit den in diesem Modul behandelten Technologien ein funktionstüchtiges, browserbasiertes Frontend für das vorgegebene Backend, welches die nachfolgenden Anwendungsfälle und nichtfunktionalen Anforderungen erfüllt:

Hauptanwendungsfälle

Gewichtung: x3

1. Als Benutzer kann ich mir eine Übersicht aller meiner Aufgaben anzeigen lassen (List)
2. Als Benutzer kann ich mit einem Formular eine neue Aufgabe zur Liste hinzufügen (Create)
3. Als Benutzer kann ich eine bestehende Aufgabe in einem Formular bearbeiten und speichern (Update)
4. Als Benutzer kann ich eine bestehende Aufgabe löschen (Delete)

Zusätzliche Anwendungsfälle

Gewichtung: x2

5. Als unauthentifzierter Benutzer kann ich mich mittels Benutzername und Passwort anmelden
6. Als authentifzierter Benutzer kann ich meine eigenen Aufgaben anzeigen, bearbeiten und löschen, unauthentifzierter Benutzer haben keinen Zugriff
7. Als Benutzer kann ich direkt auf eine Aufgabe zugreifen, indem ich einen eindeutigen URL aufrufe (Routing)
8. Als Benutzer kann ich bei fehlerhaften Eingaben gewarnt (Validation)
9. Als Benutzer werden mir Fehler auf eine benutzerfreundliche Art angezeigt (Errorhandling)

Nichtfunktionale Anforderungen

Gewichtung x1

10. Das Projekt ist sauber aufgesetzt und ist in einer sinnvollen Ordnerstruktur organisiert.
11. Der Quellcode des Projektes wird mit git Versioniert
12. Die Einhaltung von Coderichtlinien wird überprüft und sichergestellt
13. Die Funktionsweise der Applikation wird mit Tests überprüft
14. Die Applikation ist benutzerfreundlich und ansprechend gestaltet

Bewertung

Hauptanwendungsfälle

Beurteilungskriterien	Punkte			
	0	1	2	3
Gewichtung: 3x	--	-	+	++
1. Als Benutzer kann ich mir eine Übersicht aller meiner Aufgaben anzeigen lassen. <ul style="list-style-type: none"> › Daten werden als JSON mit einem GET-Request von der API abgefragt › Daten werden als Liste oder Tabelle übersichtlich dargestellt › Informationen sind sinnvoll aufbereitet 				
2. Als Benutzer kann ich mit einem Formular eine neue Aufgabe zur Liste hinzufügen. <ul style="list-style-type: none"> › Daten als JSON mit einem POST-Request an die API gesendet › Formular ist vollständig und inkl. Labels › Formular ist semantisch korrekt und es werden die richtigen <input>-Elemente verwendet 				
3. Als Benutzer kann ich eine bestehende Aufgabe in einem Formular bearbeiten und speichern. <ul style="list-style-type: none"> › Daten als JSON mit einem PUT- oder PATCH-Request an die API gesendet › Aktion kann sowohl aus der Übersicht als auch aus der Einzelansicht aufgerufen werden › Formular ist vollständig und semantisch korrekt 				
4. Als Benutzer kann ich eine bestehende Aufgabe löschen <ul style="list-style-type: none"> › Anfrage mit einem DELETE-Request an die API gesendet › Aktion kann sowohl aus der Übersicht als auch aus der Einzelansicht aufgerufen werden › Im Erfolgsfall wird eine entsprechende Meldung angezeigt 				
Anzahl pro Gütestufe				
Erreichte Punkte (max. 36 Punkte)				

Zusätzliche Anwendungsfälle

Beurteilungskriterien	Punkte			
	0	1	2	3
Gewichtung: 2x	--	-	+	++
5. Als unauthentifizierter Benutzer kann ich mich mittels E-Mail und Passwort anmelden. <ul style="list-style-type: none"> › Daten als JSON mit einem POST-Request an die API gesendet › Formular ist vollständig und semantisch korrekt › Token aus der Response wird für nachfolgende Requests gespeichert 				
6. Als authentifizierter Benutzer kann ich meine eigenen Aufgaben anzeigen, bearbeiten und löschen, unauthentifizierter Benutzer haben keinen Zugriff <ul style="list-style-type: none"> › Nach der Anmeldung wird das Token bei Requests korrekt als Header mitgeschickt › Unauthentifizierte Benutzer werden mit einer Meldung auf das Login weitergeleitet › Dem Benutzer wird angezeigt, ob er eingeloggt ist oder nicht 				
7. Als Benutzer kann ich direkt auf eine Aufgabe zugreifen, indem ich einen eindeutigen URL aufrufe <ul style="list-style-type: none"> › Die Übersicht der Aufgaben ist via URL direkt aufrufbar › Die Anzeige einer einzelnen Aufgabe ist via URL und ID-Parameter direkt aufrufbar › Das Bearbeitungsformular einer einzelnen Aufgabe ist via URL und ID-Parameter direkt aufrufbar 				
8. Als Benutzer kann ich bei fehlerhaften Eingaben gewarnt <ul style="list-style-type: none"> › Die Applikation akzeptiert keine Leerwerte für Aufgabennamen › Die Applikation akzeptiert nur korrekte E-Mail-Adressen (Regex) › Es wird dem Benutzer für jeden Fehler eine sinnvolle Fehlermeldung angezeigt 				
9. Als Benutzer werden mir Fehler auf eine benutzerfreundliche Weise angezeigt <ul style="list-style-type: none"> › Falsche oder nichtexistierende URLs werden abgefangen (z.B. mit einer Meldung oder Weiterleitung) › Serverfehler (4XX, 5XX) werden abgefangen › Beim speichern (Erstellen, Bearbeiten, Löschen) einer Aufgabe wird dem Benutzer im Erfolgsfall ebenfalls eine entsprechende Meldung angezeigt 				
Anzahl pro Gütestufe				
Erreichte Punkte (max. 30 Punkte)				

Nichtfunktionale Anforderungen

Beurteilungskriterien	Punkte			
	0	1	2	3
Gewichtung: 1x	--	-	+	++
10. Das Projekt ist sauber aufgesetzt und ist in einer sinnvollen Ordnerstruktur organisiert. <ul style="list-style-type: none"> › Die Ordnerstruktur ist zweckmässig, die Dateien sind entsprechend benannt und eingeordnet › Die Applikation verfügt über eine package.json-Datei mit den relevanten Informationen und allen nötigen Abhängigkeiten › Die Applikation verfügt über eine README-Datei mit allen wichtigen Informationen inkl. der Dokumentation der Entwicklungsumgebung 				
11. Der Quellcode des Projektes wird mit mit git versioniert <ul style="list-style-type: none"> › Git ist im Quellcodeordner korrekt aufgesetzt › Pro Halbtag des Kurses gibt es mindestens einen Commit › Commit-Messages sind aussagekräftig und sinnvoll 				
12. Die Einhaltung von Coderichtlinien wird überprüft und sichergestellt <ul style="list-style-type: none"> › Für das Projekt ist ein geeignetes Werkzeug zur automatisierten Überprüfung von Coderichtlinien eingerichtet › Das Werkzeug zeigt keine Warnungen oder Fehler an › Die gewählten Richtlinien und Ausnahmen sind sinnvoll und begründbar 				
13. Die Funktionsweise der Applikation wird mit Tests überprüft <ul style="list-style-type: none"> › Testfälle für die wichtigsten Anwendungsfälle wurden definiert (automatisiert und/oder manuell) › Testfälle für die zusätzlichen Anwendungsfälle wurden definiert › Die Testfälle laufen fehlerfrei durch 				
14. Die Applikation ist benutzerfreundlich und ansprechend gestaltet <ul style="list-style-type: none"> › Ein (ansprechendes) Design wurde mit CSS o.Ä. implementiert › Die Applikation wurde benutzerfreundlich gestaltet › Grafiken wurden zur visuellen Unterstützung eingebunden 				
Anzahl pro Gütestufe				
Erreichte Punkte (max. 15 Punkte)				