Leistungsbeurteilung 294 V2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Modulnummer | 294 | Verordnung | 2021 | Anbieter Version | 1.0 |
| Titel  Autor | Frontend einer interaktiven Webapplikation realisieren  Christian Witschard (christian.witschard@bbbaden.ch)/Manuel Bachofner | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Datum: |  |
|  |  |  |  |
| Name: |  | Vorname: |  |
| Klasse: |  | Experte: |  |
| Note: |  |

|  |  |  |
| --- | --- | --- |
| Hilfsmittel | Folgende Hilfsmittel sind erlaubt:  Alle Unterlagen  Folgende Hilfsmittel sind nicht erlaubt:  Kommunikationsmittel, wie Mobiltelefon, Messenger | |
| Prüfungsdauer | 10 Lektionen zu 45 min | |
| Prüfungsinfrastruktur | Folgende Infrastruktur steht zur Verfügung:  Eigenes Notebook | |
| Notenskala | Linear (Note = 1 + 5 \* ), auf halbe Noten gerundet.  1 ist die schlechteste, 6 die beste Note. | |
| Objekt | Front-End einer interaktiven Webapplikation. Entsprechende Darstellung für die Verwaltung (Create, Read, Update, Delete) von Daten. Zum Beispiel: Verwaltung einer Todo-Liste. |

|  |  |  |
| --- | --- | --- |
| HZ | Handlungsziel | Aufgabe(n) |
| 1 | Richtet die lokale Entwicklungs- und Laufzeitumgebung so ein, dass ein vorgegebenes Projekt entwickelt werden kann. | 1 |
| 2 | Programmiert mittels vorgegebener Technologie und mit Hilfe eines existierenden, dokumentierten Back-Ends ein effizientes, strukturiertes Front-End einer interaktiven Webapplikation, welches die Verwaltung (Create, Read, Update, Delete) von Daten ermöglicht und hält sich dabei an relevante Vorgaben. | 3 |
| 3 | Programmiert das Front-End einer interaktiven Webapplikation so, dass die einzelnen CRUD-Elemente des Front-Ends über einen permanenten Link erreichbar sind. | 3 |
| 4 | Überprüft Zwischenergebnisse mit den Anforderungen (funktional, nichtfunktional, Sicherheit) und nimmt laufend Korrekturen vor. | 2, 3,4 |
| 5 | Hält vorgegebene Coderichtlinien ein und überprüft laufend deren Einhaltung. | 1, 3 |
| 6 | Legt Änderungen und Erweiterungen der Implementierung übersichtlich und zuverlässig in einem Softwareverwaltungssystem ab. | 3 |
| 7 | Schützt mindestens einen Bereich des Front-Ends vor anonymen Zugriffen. | 3 |

Bewertung:

|  |  |  |  |
| --- | --- | --- | --- |
| Aufgabe | Gewichtung | Punkte | Bewertung |
| 1-1 | 1 | 3 | Die Konfiguration der IDE ist dokumentiert. |
| 2-1 | 1 | 3 | Es sind 3 Testfälle nach Vorgabe spezifiziert. |
| 3-1 | 1 | 3 | Die CRUD-Operation sind im Frontend umgesetzt. |
| 3-2 | 1 | 3 | Die CRUD-Operationen werden an das Web-API weitergeleitet. |
| 3-3 | 1 | 3 | Die Formulardaten sind clientseitig mit JavaScript validiert. |
| 3-4 | 2 | 3 | Die Software ist professionell implementiert  (Coding Standards, Module). |
| 3-5 | 1 | 3 | Das Client-Side Routing ist für mindestens drei Seiten umgesetzt |
| 3-6 | 1 | 3 | Sinnvolle Zwischenstände sind im GIT Repository ersichtlich. |
| 3-7 | 1 | 3 | Der Benutzer kann sich über eine Web API authentifizieren. |
| 3-8 | 1 | 3 | Die Autorisierung für einen Bereich der Webseite ist professionell umgesetzt. |
| 4-1 | 1 | 3 | Die spezifizierten Testfälle sind automatisiert umgesetzt. |
|  | 1 | 3 | Das Projekt ist lauffähig und die administrativen Vorgaben sind eingehalten. |

**Hinweise**:

* Diese Prüfung ist eine Einzelarbeit.
* Es dürfen keine alten Prüfungen und/oder Daten von alten Prüfungen verwendet werden.
* Betrügereien, wie Spicken o.Ä., oder Betrugsversuche führen zum Ausschluss aus der Prüfung.
* Nicht deklarierte Kopien von Codeschnipseln aus dem Internet gilt als Betrug.
* Im Falle eines Betrugs gilt die Leistungsbeurteilung als absolviert, aber nicht bestanden.
* Geben Sie diese Aufgabenstellung und den Lösungsbogen mit Nachname, Vorname und Klasse beschriftet ab.
* Bezeichnen Sie Lösungen und Korrekturen klar und deutlich.
* Verwenden Sie einen dokumentenechten Stift (Tinte/Kugelschreiber).
* Wenn Sie Lösungen elektronisch oder auf Zusatzblättern abgeben, verweisen Sie auf diese Zusatzmaterialien auf Ihrem Aufgabenblatt.
* Alle Dateien müssen in einem Ordner abgelegt werden, dessen Name dem Format NameVornameLB-294 entspricht. (z.B. MuellerHansLB-294).
  + Der Ordner muss vor der Abgabe gepackt werden.
  + Die gepackte Datei muss gleich wie der Order benannt werden.
  + Programmierprojekte müssen gleich wie der Ordner benannt werden.
* Folgende Formate sind bei den elektronischen Lösungen erlaubt:
  + Programmcode: Visual Studio-Projekte oder Dateien mit Ordnerhierarchie
  + Bilder: jpg, gif, png
  + Texte: ASCII-Texte, Microsoft Word, Adobe Acrobat pdf
  + Anderes: Microsoft Excel, Microsoft PowerPoint
* Die elektronischen Lösungen werden am Schluss der Prüfung auf Moodle hochgeladen.
* Elektronisch abgegebene Lösungen werden mit Vorteil gezippt.
* Sie sind verantwortlich dafür, dass Sie alle Dateien abgeben.
* Alle Daten dieser LB (Aufgabendateien und Lösungen) müssen sofort nach dem Abschluss der LB und dem Hochladen gelöscht werden.
* Vor Beginn der Prüfung abgegebene Daten:
  + Keine oder Auflistung der Dateien
* Regelung zu verspätet eingereichter Leistungsbeurteilungen:
  + Pro angefangene 24h Verspätung wird eine halbe Note (0.5 Notenpunkte) abgezogen. Verspätet eingereichte Leistungsbeurteilungen werden korrigiert, bewertet und dann mit dem Abzug der Verspätung verrechnet.
  + Aufschiebende Wirkungen haben nicht planbare Absenzen wie Krankheit oder Unfall, sofern Sie vom Berufslernenden bei Eintritt des Ereignisses kommuniziert wurden. Die Lehrperson entscheidet über die Abgabetermine.

Administrative Vorgaben

Halten Sie alle administrativen Vorgaben (unter anderem diejenigen auf Seite 4) ein.

Um Ihre Lösungen elektronisch abzugeben, erstellen Sie ein Word-Dokument mit dem Namen: <<NachnameVorname\_LB\_M294\_V2.docx>>. In diesem Dokument dokumentieren Sie das Projekt. Alle Bilder und Skizzen müssen elektronisch in diesem Dokument lesbar erfasst werden.

Form und Ablauf der Leistungsbeurteilung

Sie bearbeiten im Rahmen des Moduls 294 eine kleine Projektarbeit. Sie führen diese Arbeiten allein durch – fremde Hilfe und kopierte Codeteile aus dem Internet müssen Sie mit Quellangabe deklarieren. Bewertungsrelevante Teile der Prüfung müssen Sie selbst erstellen.

Sie erhalten Rahmenbedingungen in Form von Anforderungen. Diese ergänzen Sie mit den eigenen Ideen zu einem kleinen Pflichtenheft.

Nach Genehmigung des Pflichtenhefts setzen Sie das Projekt gemäss dieser Aufgabenstellung um.

Situation

Sie erstellen mit HTML5 und JavaScript eine kleine Webanwendung als Single Page Applikation. Dabei können Sie das Thema selbst wählen, die Rahmenbedingungen sind aber vorgegeben. Falls Sie die SPA stylen wollen, dürfen Sie CSS3 verwenden.

**Funktionale Anforderungen**

1. Die Webanwendung besitzt einen Header, einen Footer, einen Navigationsbereich und einen Hauptanzeigebereich.
2. Die Webanwendung besitzt mindestens 3 navigierbare Anzeigen.
3. Die Webanwendung stellt Daten aus einem REST API an.
4. Mit der Webanwendung sollen die Daten mittels CRUD verändert werden können.
5. Die Eingabedaten müssen clientseitig validiert werden.
6. Die Manipulation von Daten muss durch ein Login geschützt werden.
7. Auf jeder einzelnen Seite müssen der Name und Vorname des Erstellers ersichtlich sein.

**Qualitätsanforderungen**

1. Die Webseite muss automatisch getestet werden.
2. Gängige Coderichtlinien müssen eingehalten werden.
3. Die Entwicklung soll in einem GIT Repository nachvollziehbar abgebildet werden.
4. Es soll das Client Side Routing verwendet werden.

**Randbedingungen**

1. Dokumentation in einer Worddatei.
2. Erstellung mit HTML5, CSS3 und JavaScript
3. Abgabe gemäss Angaben Lehrperson: Projekt voll funktionsfähig und direkt zu öffnen in Visual Studio Code, weitere Unterlagen gemäss administrativen Angaben.
4. Die Webseite muss nur auf der aktuellen Version von Edge getestet werden.
5. Die Webseite muss mit dem abgegebenen REST API funktionieren.
6. Die Webseite soll eine Liste von Datensätzen abbilden.
7. Ein Datensatz soll mindestens 4 Eigenschaften mit verschiedenen Datentypen beinhalten.

Prüfungsaufgaben

***Aufgabe 0: Projektbeschrieb (Vorarbeit - nicht bewertet)***

Sie erstellen auf der Basis der geforderten Anforderungen einen kurzen Projektbeschrieb für die Webapplikation. Die Anforderungen sind bereits hier gegeben. Diese müssen Sie nicht nochmals auflisten. Der Beschrieb soll nur Text enthalten und keine Bilder. Im Umfang sollte er maximal eine Seite lang sein.

Achten Sie darauf, dass Ihr Projekt die geforderten Elemente enthält. Der Projektbeschrieb soll eine Idee der Webapplikation geben bezüglich des Inhalts.

Die Applikation soll eine Liste von Datensätzen abbilden.

Denken Sie sich nicht zu komplexe Anforderungen für die Anwendungslogik aus.

Erstellen Sie auf Grund der Anforderungen eine JSON Schnittstellendokumentation.

Den Projektbeschrieb und die JSON Dokumentation senden Sie an die Lehrperson. Diese gibt das Projekt frei oder fordert noch Änderungen an. Die Umsetzung kann beginnen, wenn das Projekt genehmigt ist.

Als Backend verwenden Sie die im Modul bereitgestellte Lösung. Sollten Sie eine andere Lösung verwenden wollen, sprechen Sie dies mit ihrer Lehrperson ab.

Projektbeispiele:

* Inventar von Tieren einer Zoohandlung
* Einfache Adressliste
* Installierte Software auf dem PC

***Aufgabe 1: IDE***

Schreiben Sie auf, welche IDE Sie verwenden.

Notieren Sie dazu die Plugins/Erweiterungen, welche Sie für die Umsetzung des Projekts verwendet haben. Schreiben Sie bei jedem Plugin den Verwendungszeck.

IDE: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Plugins:

|  |  |
| --- | --- |
| **Plugin/Erweiterung** | **Verwendungszweck** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

***Aufgabe 2: Testfallspezifikation***

Schreiben Sie eine Testfallspezifikation mit mindestens drei Testfälle für maximal zwei Anforderungen.

***Aufgabe 3: Umsetzung der SPA***

Programmieren Sie nun Ihre Single Page Applikation nach Vorgabe. Halten Sie dabei die gängigen Coderichtlinien ein. Vorgehen:

* Erstellen Sie auf git.bbbaden.ch ein Repository. In diesem speichern Sie Ihre Codestände nach Vorgabe mit Kommentaren ab.
* Erstellen Sie nun die einzelnen Elemente der Applikation
* Erstellen Sie eine Anzeige mit einer Datensatzübersicht
* Erstellen Sie eine Detailansicht eines einzelnen Datensatzes
* Erstellen Sie eine Login Seite, welche die Datenmanipulation schützt.
* Erstellen Sie eine Seite zum Verändern eines Datensatzes
* Erstellen Sie eine Funktion zum Löschen eines Datensatzes

***Aufgabe 4: Testing***

Setzen Sie die in Aufgabe 3 spezifizierten Tests mit dem Selenium Plugin um. Exportieren Sie die Tests.

Führen Sie die Tests durch und dokumentieren Sie die Ergebnisse in einem Testprotokoll und Testfazit.

***Aufgabe 5: Abgabe***

Folgende Elemente müssen zur Bewertung auf Moodle abgegeben werden:

* Eine ZIP-Datei von maximal 64MB mit:
* der Dokumentation
* dem erstellten Code
* den erstellten Testskripten

Geben Sie das GIT Repository für die Lehrperson frei.

Sie halten das Abgabedatum/ die Abgabezeit gemäss Angaben der Lehrperson ein.

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst und keine anderen Hilfsmittel als die angegebenen verwendet habe.

Insbesondere versichere ich, dass ich alle wörtlichen und sinngemäßen Übernahmen aus anderen Werken als solche kenntlich gemacht habe.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Ort, Datum)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Unterschrift)