

# Meilenstein 6 – E-Mensa (Laravel)

Abgabe: 13.01.2023

Ziel ist die Entwicklung einer mobilen Webseite, mit der Benutzer:innen einzelne Gerichte bewerten können. Dank der mobilen Webseite können Benutzer:innen direkt nach dem Verzehr ihre Meinung über das von ihnen ausgewählte Gericht mit anderen teilen.

Sie haben die Wahl: Sie können die Inhalte dieses Meilensteins ...

- a) mit dem Web Application Framework Laravel in einem neuen Projekt (neues Verzeichnis /emensamobil) umsetzen oder
- b) das Zwischenergebnis (/emensa) der letzten Meilensteine 4 & 5 weiterentwickeln.

Sollten Sie das Review mit einem anderen Team in Meilenstein 4 durchgeführt haben, so können Sie eine der mit (Review-Optional) markierten Aufgaben als (Optional) ansehen.

## Aufgabe 1

E-Mensa: Bewertung von Gerichten. Ihr nächster Auftrag der E-Mensa GmbH ist die Entwicklung eines weiteren Features für das Projekt: Alle registrierten Nutzer:innen sollen eine Bewertung zu den Gerichten abgeben können. Die aktuellen Bewertungen sind für andere Nutzer:innen sichtbar.

Sie erhalten die folgende Beschreibung der E-Mensa GmbH:

*„Angemeldete Benutzer:innen können mehrere Bewertungen pro Gericht abgeben. Eine Bewertung besitzt eine notwendige Bemerkung (mit einer Mindestlänge von 5 Zeichen), eine ‚Sterne-Bewertung‘ mit vier Optionen (sehr gut, gut, schlecht, sehr schlecht) sowie einen Bewertungszeitpunkt. Der Bewertungszeitpunkt ist stets der Zeitpunkt, an dem die Bewertung in der Datenbank eingetragen wird. Außerdem kann eine Bewertung durch eine Person mit Administrationsrechten hervorgehoben werden. Eine hervorgehobene Bewertung erscheint auf der Hauptseite.“*

- a) Entwerfen Sie das ER-Model (in Chen-Notation) in elektronischer Form für die nötigen Änderungen an der Datenbank. Falls möglich, erweitern Sie Ihr bisheriges ER-Model. Speichern Sie das ER-Model als Bild unter **/beispiele/m6\_a1.{png, jpeg, jpg}** ab.  
*Hinweis: Bilden Sie die Optionen der Sterne-Bewertung nicht als eigenen Entitätstypen ab.*
- b) Setzen Sie die Änderungen an der Datenbank per SQL um und speichern Sie die nötigen Befehle in der Datei **/beispiele/m6\_a1.sql** ab.

- c) Ihre Wahl: Implementieren Sie die Erweiterung in `/emensa` oder setzen Sie das Projekt mit Laravel (unter `/emensamobil`) neu auf, wobei Sie die bestehende Datenbank weiterverwenden.

Berücksichtigen Sie bei der Implementierung die folgenden weiteren Anforderungen:

- 1) Das Formular für die Bewertungseingabe soll unter `/bewertung` dargestellt werden. Das Formular steht nur angemeldeten Benutzer:innen zur Verfügung. Nicht angemeldete Benutzer:innen werden automatisch zur Anmeldung weitergeleitet. Nach der erfolgreichen Anmeldung werden Nutzer:innen wieder zu `/bewertung` automatisch weitergeleitet.
- 2) Implementieren Sie eine sinnvolle Prüfung der Eingabewerte des Bewertungsformulars und speichern Sie die Daten in der Datenbank.
- 3) Auf der Hauptseite soll für jedes gezeigte Gericht ein Link zur Bewertung angeboten werden. Der Link führt zu `/bewertung` und enthält die Id des Gerichts, welches bewertet werden soll (z.B. `/bewertung?gerichtid=42`). Nur angemeldete Benutzer:innen sehen den Link.
- 4) Bei dem Bewertungsformular wird der Name sowie das zugehörige gespeicherte Bild des zu bewertenden Gerichts dargestellt.
- 5) (Review-Optional) Das Bewertungsformular soll sich auf kleinen Displays anders verhalten als auf großen. Zeigen Sie bei einer Breite des Viewports (z.B. unter 600px) die Labels und Eingabeelemente alle untereinander an. Bei einer größeren Breite sollen die Labels links neben den Eingabeelementen positioniert sein. Nutzen Sie dafür CSS Media-Queries, CSS Flexbox oder eine passende Bibliothek, wie Bootstrap.
- 6) Zeigen Sie unter `/bewertungen` die letzten 30 erfassten Bewertungen an. Zeigen Sie die Bewertungen in chronologisch absteigender Reihenfolge an. Fügen Sie zum Testen mindestens 10 Bewertungen Ihrer Wahl ein.
- 7) Zeigen Sie angemeldeten Benutzer:innen unter `/meinebewertungen` alle selbst erfassten Bewertungen (dieser Benutzer:in) an. Zeigen Sie die Bewertungen in chronologisch absteigender Reihenfolge an.
- 8) Implementieren Sie ein Löschen von Bewertungen. Dabei können Benutzer:innen nur ihre eigenen Bewertungen löschen. Benutzer:innen können nicht die Bewertungen anderer Benutzer:innen löschen. Bieten Sie in der Liste der Bewertungen den Link zum Löschen an.
- 9) (Optional) Bieten Sie unter `/bewertungen` eine Filterung nach Gerichten für die Bewertungen an.

## Aufgabe 2

E-Mensa: Bewertungen hervorheben. Die E-Mensa GmbH möchte einige der Bewertungen auf der Hauptseite der E-Mensa darstellen, damit neue Besucher:innen schnell einen Eindruck von der Qualität der Gerichte erhalten. Dafür sollen Benutzer:innen mit der Rolle „Administrator“ der E-Mensa gezielt erfasste Bewertungen aussuchen und für die Darstellung auf der Hauptseite hervorheben können.

Implementieren Sie dazu die folgenden Anforderungen:

- 1) Über die neue Darstellung der Bewertungen unter `/bewertungen` sollen Administratoren einzelne Bewertungen hervorheben können. Ein/e Administrator:in führt eine Hervorhebung durch einen Klick auf einen Link „Hervorheben“ durch, der pro Bewertung angeboten wird. Die Hervorhebung soll in der Datenbank gespeichert sein. Administrator:innen erkennt man am Attribut *admin* der Tabelle **benutzer** (*admin=true*).
- 2) Administrator:innen können die Hervorhebung bei hervorgehobenen Bewertungen durch einen Klick auf einen Link „Hervorhebung abwählen“ wieder zurücknehmen.
- 3) Eine hervorgehobene Bewertung soll auch visuell hervorgehoben unter `/bewertungen` dargestellt werden. Am besten erreichen Sie dies, indem Sie eine CSS-Klasse mit einem passenden Styling dafür verwenden.
- 4) (Review-Optional) Hervorgehobene Bewertungen erscheinen auf der Startseite in einem neuen Bereich „Meinungen unserer Gäste“ untereinander. Zeigen Sie den Namen des bewerteten Gerichts, die Bemerkung sowie die Bewertung an.

## Woche 2

Empfehlung VL: ORM

## Aufgabe 3

E-Mensa: Optimierung durch Einsatz eines Objekt-Relationales Mappings (ORM). Wir verbessern unsere bisherige Lösung, indem wir einige der verwendeten SQL-Abfrage durch eine objektorientierte Schreibweise abkürzen. Dafür verwenden wir Eloquent.

- 1) Installieren Sie das Paket zu Eloquent mit Namen illuminate/database<sup>8</sup> über den Composer.

```
php bin/composer.phar require illuminate/database
```

*Hinweis: In Laravel ist Eloquent bereits installiert und dieser Schritt ist nicht notwendig.*

- 2) Kopieren Sie die mitgelieferte **orm.php** in Ihr Projekt nach **config/orm.php** und inkludieren Sie in der **public/index.php** die neue Konfigurationsdatei.  
*Hinweis: In Laravel ist dieser Schritt nicht notwendig.*
- 3) Verwenden Sie ein Model von Eloquent zum Laden und Speichern einer „Bewertung“. Verwenden Sie das Model für das ...
  - a) Auslesen des Namens des Gerichts unter /bewertung
  - b) Aktualisieren der Hervorhebung unter /bewertungen
  - c) Löschen von Bewertungen unter /meinebewertungen
- 4) Schreiben Sie ein Model für ein „Gericht“. Verwenden Sie dieses Model an mindestens einer weiteren Stelle im Projekt E-Mensa.
- 5) Sorgen Sie dafür, dass die Attribute *preis\_intern* und *preis\_extern* beim Model „Gericht“ beim Zugriff aus der Anwendung heraus stets mit zwei Nachkommastellen zur Verfügung stehen. Verwenden Sie Accessors / Mutators dafür.
- 6) Sorgen Sie dafür, dass bei den Attributen *vegetarisch* und *vegan* auch die Werte „Yes“, „Ja“ (für wahr) und „No“, „Nein“ (für false) gesetzt werden können und korrekt (intern als: wahr oder falsch) in die Datenbank geschrieben werden. Eine beliebige Groß- und Kleinschreibung sowie beliebige Leerzeichen in den Werten, wie „Y e S“ oder „n O“, sind zulässig und sollen intern korrekt umgesetzt und gespeichert werden.

<sup>8</sup> Verfügbar unter <https://packagist.org/packages/illuminate/database> (Letzter Zugriff 20.9.2022)

## Aufgabe

Abgabe. Sie geben Ihre erarbeiteten Ergebnisse des Meilensteins ab.

- 1) Aktualisieren Sie Ihr Dossier.
- 2) Prüfen Sie im PTV, ob Ihre vorherige erfolgreiche Abnahme eingetragen ist.
- 3) Laden Sie Ihre Ergebnisse als ZIP in ILIAS im Abgabeordner des Meilensteins im Format wie beim letzten Meilenstein hoch.

## Abschluss

Feier. Nach Fertigstellung und Abnahme Ihrer letzten Aufgabe bei der E-Mensa GmbH dankt Ihnen das Unternehmen herzlich für Ihr Engagement und das Arbeitsergebnis. Sie denken an Ihre gelernten und vertieften Kenntnisse und Fähigkeiten während der gemeinsamen Einarbeitung im Team zurück:

- **HTML** und **CSS** einsetzen
- Dynamische Inhalte mit der Programmiersprache **PHP** serverseitig erzeugen
- Das Aufsetzen und Inbetriebnehmen einer **Datenbank**
- **SQL** einsetzen und mit der Datenbank **MariaDB** arbeiten
- Anforderungstexte als **ERD nach Chen** erfassen und weiterentwickeln
- Umsetzung von Anforderungen in einer **Webanwendung** nach dem **MVC**-Muster mit einer modernen Entwicklungsumgebung (wie PHPStorm)

**Herzlichen Glückwunsch** zum Abschluss des letzten Meilensteins des Praktikums von Datenbanken und Webtechnologien!

Wir wünschen Ihnen **viel Erfolg** und **gutes Gelingen** bei der Klausur sowie bei Ihren zukünftigen Umsetzungen von Webanwendungen!