

# LAP-Abschlussprüfung

Beachten Sie bitte folgende Punkte:

1. Speichern Sie Ihre Daten während der Prüfung ausschließlich lokal ab – erst bei Abgabe speichern Sie Ihre Daten am Q-Laufwerk im Order »ABGABE« ab.
2. Der Austausch von Daten untereinander sowie jedwelche Unterstützung durch/für andere Beteiligte ist untersagt.
3. Der Zugriff auf das Internet sowie unser Skriptum bzw. ein anderes Skriptum ist erlaubt.
4. Im Teil 3 müssen Sie keinen Wert auf Formatierung legen.
5. Die Prüfungszeit ist wie folgt:
  1. Teile 1+2: 3.5 Stunden
  2. Teil 3: 4 Stunden
6. Die Pauseneinteilung erfolgt nach Ihrem Ermessen (einzeln)

## Abgabe

Geben Sie die Teile 1-3 getrennt in Form einzelner ZIP-Dateien ab. Legen Sie sich hierzu im Verzeichnis ABGABE einen eigenen Ordner, benannt mit Nachname\_Vorname an und speichern Sie dort Ihre ZIP-Dateien, benannt mit Teil1.zip, Teil2.zip und Teil3.zip.

# Teil 1 – Datenbankerstellung

Mit \* gekennzeichnete Felder sind Pflichtfelder.

Dokumentieren Sie die Datenbank mithilfe eines ER-Diagramms und einer schriftlichen Dokumentation wie bei der LAP gefordert. Dokumentieren Sie ebenso die Constraints.

## Aufgabe

Erstellen Sie ein Datenbankmodell in der dritten Normalform für untenstehendes Beispiel. Entwerfen Sie eine Datenbank für die Speicherung von Lego-Modellen samt Bauteillisten sowie den Bauteile-Bestand von Usern. Hierbei sollen die einzelnen Lego-Bauteile nach Farbe, Form und Größe abgelegt werden.

Eine Bauteilliste wird immer einem Modell zugewiesen. Beispiel:

- Modell 23 benötigt folgende Bauteile:
  - 7 graue Plättchen Teilung 2x3
  - 5 blaue Plättchen Teilung 1x4
  - 8 rote Steine Teilung 2x2
  - usw.

Beachten Sie, dass unterschiedliche Modelle (zu unterschiedlichen Zeiten) ggf. aus denselben Bauteilen gebaut werden können. Außerdem sind die Bauteile Usern zugewiesen.

Das **Userszenario** ist also wie folgt: Ein User besitzt eine gewisse Menge an Bauteilen, die entweder durch den Kauf ganzer Modelle, durch gezielten Bauteilkauf, durch Schenkung oä. zustande gekommen ist. Außerdem existieren Baupläne von Modellen, wo eine Bauteilliste (»Stückliste«) vorhanden ist. Der User möchte also all seine Bauteile speichern können. Die Bauteillisten von Modellen sind »global« und gehören keinem User. Es soll jedoch möglich sein, dass ein User die Bauteilliste eines noch nicht im System vorhandenen Modells eingeben kann. Dass diese Eingabe dann von einem User erfolgt ist, soll im System abgebildet sein (»User XY hat die Bauteilliste von Modell 23 angelegt«); genauso kann es aber Modelle geben, die bereits im System vorhanden sind und nicht von einem User eingegeben wurden (solchen Modelle waren also bereits vorab vorhanden).

Folgende Kerndaten sollen gespeichert werden:

- User:
  - Vorname\*
  - Nachname\*
  - Geburtsdatum

- Emailadresse\* (eindeutig im System)
- Passwort\* (wird verschlüsselt gespeichert)
- Profilbild
- Modell:
  - Bezeichnung\*
  - Lego-Seriennummer (max. 20 Stellen mit Buchstaben und Ziffern laut Lego)
  - Beschreibung
  - Markteinführung (Jahr)
- Bauteilliste eines Modells:
  - Anzahl\*
  - Bauteil\* (mit allen benötigten Merkmalen)
- Bauteile:
  - Art\* (Plättchen, Stein, etc).
  - Teilung\* in x- und y-Richtung (zb. 2x4)
  - Farbe\*

\*: Pflichtfelder

Dokumentieren Sie diese Datenbank mithilfe eines ER-Diagramms und einer schriftlichen Dokumentation. Dokumentieren Sie ebenso die Constraints.

## Bausteinbeispiele

Betrachten Sie untenstehende Beispiele für Bauteile:

Legosteine (normale Höhe):

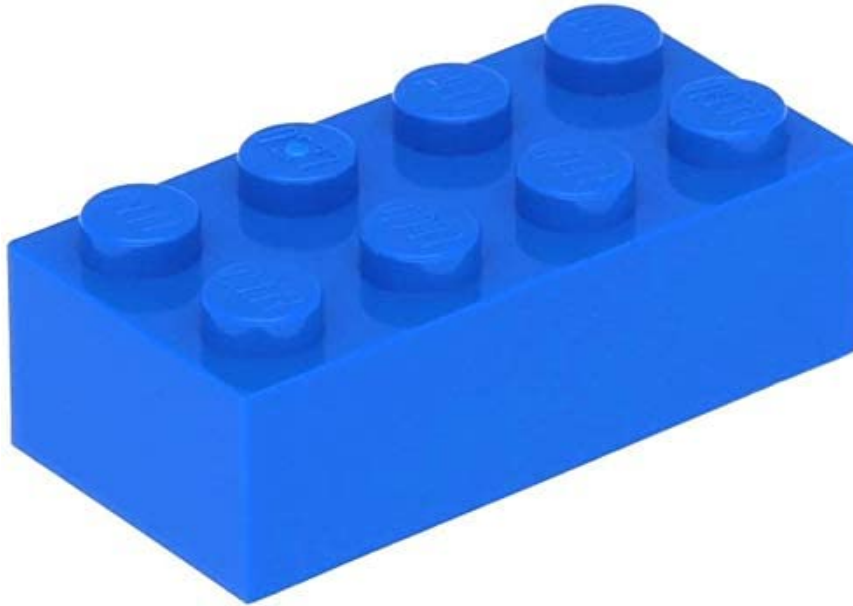


Abbildung 1: Blauer Legostein mit Teilung 2x4  
Legoplättchen (geringere Höhe):



Abbildung 2: Graues Legoplättchen Teilung 2x4

# Teil 2 – Testing

Installieren und testen Sie das Ihnen zur Verfügung gestellte Dokument. Achten Sie dabei darauf, dass Sie es zunächst korrekt (beachte: Unterverzeichnisse) auf den Server laden.

# Teil 3 – Programmierung

Importieren Sie den Ihnen zur Verfügung gestellten Datenbank-Dump in eine Datenbank Ihrer Wahl und setzen Sie die untenstehenden Seiten programmatisch um. Auf die Formatierung der Seiten muss kein Wert gelegt werden, sehr wohl jedoch auf die korrekte Semantik der HTML-Elemente.

Beachten Sie, dass von jeder Seite der Website auf eine jede andere Seite navigiert werden können soll.

## Aufgabenstellung

Verwenden Sie die Ihnen bereitgestellte Datenbank mit den bereits befüllten Tabellen. Erstellen Sie alsdann **die nachfolgenden** Ansichten. Bitte beachten Sie, dass Sie eine Startseite erstellen, von der aus man zu den einzelnen Ansichten gelangt:

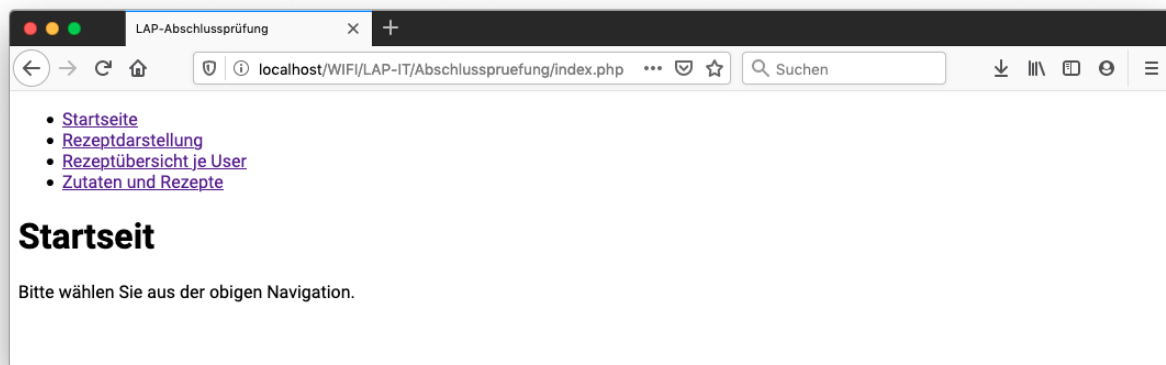


Abbildung 3: Die Startseite; auf die Formatierung brauchen Sie keinen Wert legen.

## Rezeptdarstellung

Geben Sie alle Rezepte, geordnet nach dem Titel aufsteigend, aus, und zwar in Form einzelner **<article>**. Jedes Rezept soll beinhalten:

- Titel (von Vorname Nachname)
- allgemeine Beschreibung
- Zeit für die Vorbereitung, Zeit für die Zubereitung, Zeit für das Ruhen
- Anzahl der Personen, für die das Rezept erdacht ist
- Schwierigkeitsgrad der Zubereitung
- Die benötigten Zutaten (Anzahl, Einheit, Zutat)
- Die einzelnen Zubereitungsschritte; geben Sie den Titel des Zubereitungsschrittes samt nachfolgendem Doppelpunkt nur dann aus, wenn ein Titel gespeichert ist

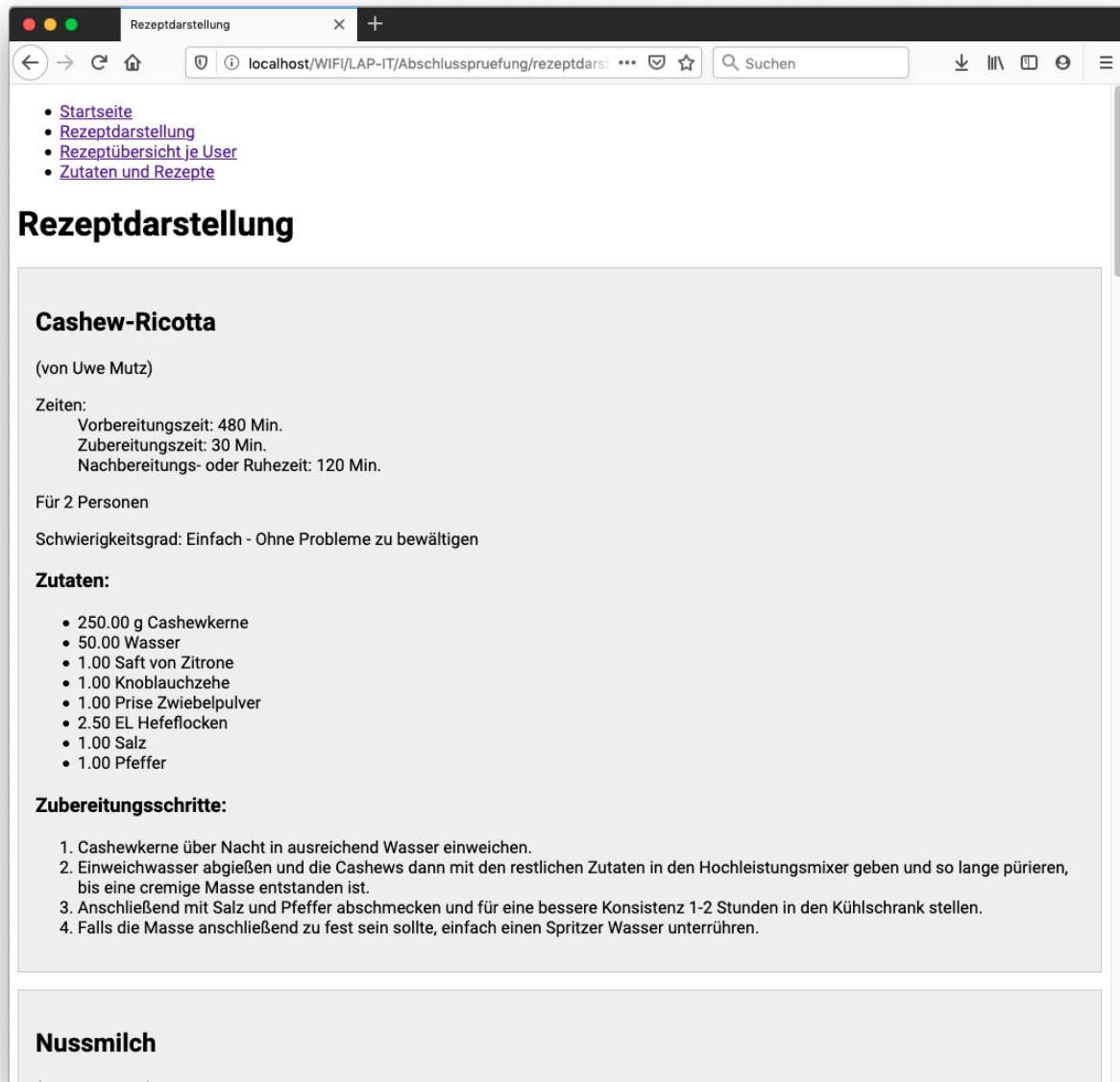


Abbildung 4: Rezeptdarstellung mit allen Einzelheiten.

## Rezeptübersicht je User

Erstellen Sie ein Formular, in welches Sie Vor- und/oder Nachnamen von Usern eingeben können. Nach Eingabe von Vor- und/oder Nachname sollen die für diese User vorhandenen Rezepte dargestellt werden, und zwar in folgender Form:

- Vorname Nachname (Emailadresse):  
Rezepttitel: Rezeptbeschreibung
- Vorname Nachname (Emailadresse):  
Rezepttitel: Rezeptbeschreibung

Bitte beachten Sie, dass die Auflistung der Personen sortiert nach Nachname und Vorname erfolgen soll und dass in beiden Fällen auch die Eingabe von Teilen des Namens möglich sein soll.

Beachten Sie auch, dass die ins Formular eingegebenen Suchkriterien (Vorname und/oder Nachname) nach Abschicken des Formulars weiterhin in den Eingabefeldern darstellt werden sollen.

Sollte kein Vor- oder Nachname eingegeben werden, so sollen alle Rezepte dargestellt werden (»leerer Filter«).

## Zutaten und Rezepte

Erstellen Sie ein Formular, in welchem alle vorhandenen Zutaten in Form einer **<select>**-Box ausgegeben werden<sup>1</sup>. Bei Auswahl einer Zutat sollen alle Rezepte, die diese Zutat nutzen, in folgender Form aufgelistet werden (sortiert nach dem Rezepttitel):

- Rezepttitel (von Vorname Nachname, für Anzahl Personen):  
Rezeptbeschreibung
- Rezepttitel (von Vorname Nachname, für Anzahl Personen):  
Rezeptbeschreibung
- ...

**Tipp:** sollten Rezepte mehrfach dargestellt werden, so verwenden Sie den »GROUP BY« **Modifier** im SQL-Statement – siehe hierzu etwa <https://dev.mysql.com/doc/refman/8.0/en/group-by-modifiers.html>.

### Ohne Filter:

Sollte keine Zutat ausgewählt worden sein, so sollen alle Rezepte ausgegeben werden.

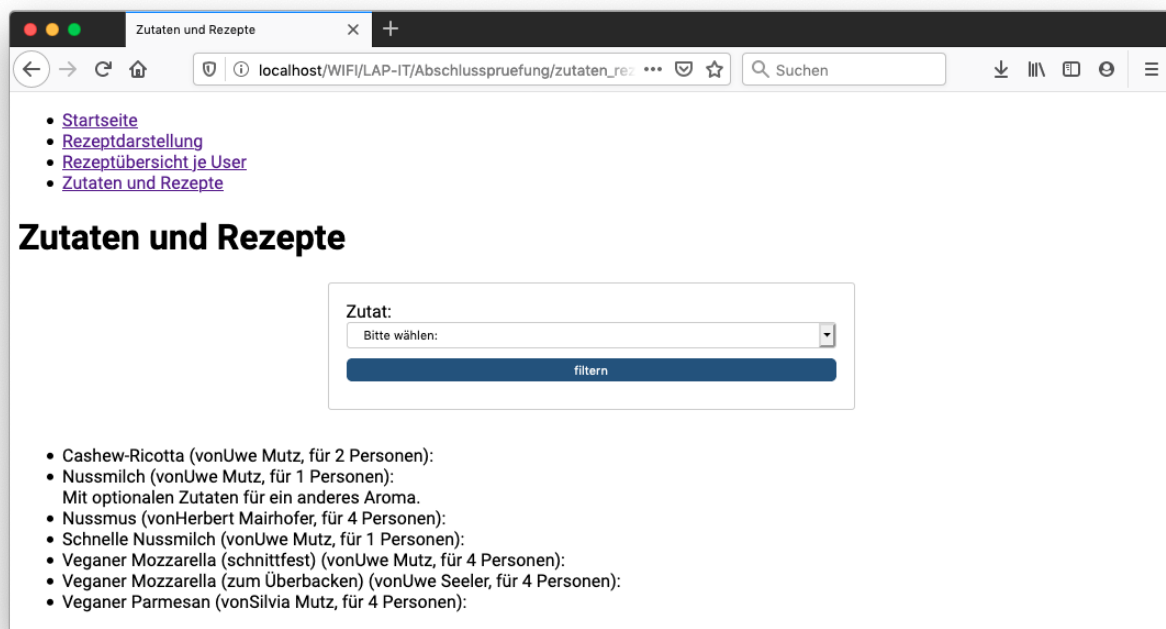


Abbildung 5: Rezeptübersicht ohne Filter

### Mit Filter:

<sup>1</sup> Sollte Ihnen das nicht gelingen, so verwenden Sie ein normales Eingabefeld, sodass Sie mit dem Beispiel fortfahren können.



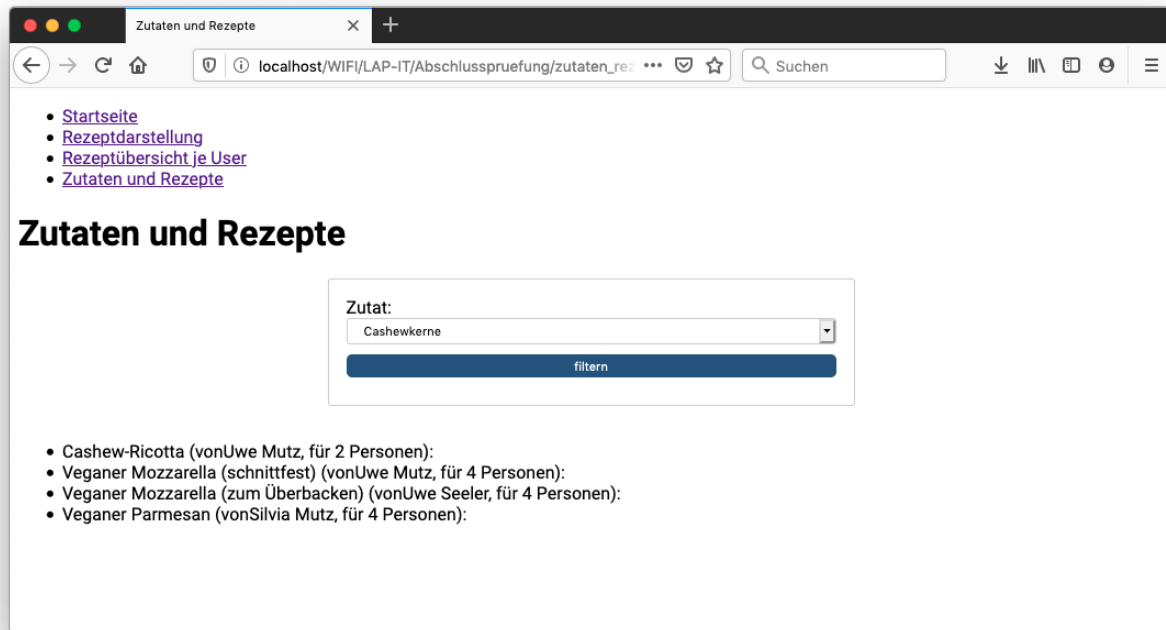


Abbildung 6: Gefiltert nach »Cashewkerne«