

Änderungen seit letzter definitions.pdf

- mehr Exceptions
- mehr Hilfsfunktionen
- priceCent, priceEuro

Table of Contents

Kursinformationen.....	2
courses.csv Struktur.....	3
Gästebucheintragsinformationen.....	4
entries.csv Struktur.....	4
Accountinformationen.....	4
users.csv Struktur.....	5
\$_SESSION Informationen.....	5
Lesen und interpretieren einer CSV Datei.....	6
Wieder-einlesbares Schreiben in eine CSV Datei.....	7
Verzeichnisstruktur.....	8
Globale Variablen.....	9
Exceptions.....	11
Hilfsfunktionen.....	12
Primärfunktionen.....	15
Effekte der Primärfunktionen.....	16
Sekundärfunktionen.....	18
Effekte der Sekundärfunktionen.....	20
Primärfunktion auslösende HTML Formulare.....	22

Kursinformationen

- Bestandteile eines Assoziativen Arrays (\$course), welches einen Kurs repräsentiert
 - \$course[„id“] = 13
- Bestandteile eines Kurses in der Datei courses.csv, mit Ausnahme von courseImage und instructorImage (Binärdaten)
- HTML Formular-namen
 - <input type=“file“ name=“courseImage“>
- HTML innerText Werte
 - <h2 class=“kursitel“> <?php echo \$course[„title“]; ?> </h2>

Deutsch	In \$_POST von Kurserstellung enthalten	In \$_POST von Kursbearbeitung enthalten	Ausliefern an Shopansicht	Ausliefern an Kursbearbeitung	Ausliefern an erstellte Kurse, gekaufte Kurse	Ausliefern an Kursansichten	Variablenname / Input name
ID	Nein	Ja	Ja	Ja	Ja	Ja	id
Titel	Ja	Ja	Ja	Ja	Ja	Ja	title
Kurzbeschreibung	Ja	Ja	Ja	Ja	Nein	Ja	summary
Kursbild (Binärdaten)	Ja	Ja	Nein	Nein	Nein	Nein	courseImage
Pfad zum Kursbild	Nein	Nein	Ja	Ja	Ja	Ja	courseImagePath
Cent-Teil des Preises	Ja	Ja	Ja	Ja	Nein	Ja	priceCent
Euro-Teil des Preises	Ja	Ja	Ja	Ja	Nein	Ja	priceEuro
Das wirst du lernen	Ja	Ja	Nein	Ja	Nein	Ja	educationalContent
Für wen ist der Kurs geeignet	Ja	Ja	Nein	Ja	Nein	Ja	suitableFor
Was ist im Kurs enthalten	Ja	Ja	Nein	Ja	Nein	Ja	materialContent

Kursersteller Name	Ja	Ja	Ja	Ja	Nein	Ja	instructorName
Kontaktmöglichkeiten	Ja	Ja	Nein	Ja	Nein	Ja	instructorContact
Bild des Kurserstellers (Binärdaten)	Ja	Ja	Nein	Nein	Nein	Nein	instructorImage
Pfad zum Bild des Kurserstellers	Nein	Nein	Nein	Ja	Nein	Ja	instructorImagePath
Kurs für Hörbehinderte geeignet	Ja	Ja	Nein	Ja	Nein	Ja	hImpairedSuitability
Kurs für Sehbehinderte geeignet	Ja	Ja	Nein	Ja	Nein	Ja	vImpairedSuitability
Alt Text für Kursbild	Ja	Ja	Ja	Ja	Ja	Ja	courseImageAlt
Menge von teilnehmenden Benutzer-IDs	Nein	Nein	Nein	Nein	Nein	Nein	participants
ID des Erstellers des Kurses	Nein	Nein	Nein	Nein	Nein	Nein	creatorId

courses.csv Struktur

- Pro Kurs zwei CSV Felder in folgender Reihenfolge

1. CSV Feld mit folgenden 16 Elementen:
<id, title, summary, courseImagePath, priceCent, priceEuro, educationalContent, suitableFor, materialContent, instructorName, instructorContact, instructorImagePath, hImpairedSuitability, vImpairedSuitability, courseImageAlt, creatorId>
2. <participants>

Gästebucheintragsinformationen

- Bestandteile eines Assoziativen Arrays (\$entry), welches einen Gästebucheintrag repräsentiert
- Bestandteile eines Gästebucheintrags in entries.csv
- HTML Formular-namen
- HTML innerText Werte

Deutsch	In \$_POST von Gästebuch enthalten	Ausliefern an Gästebuch	Variablenname / Input name
Titel	Ja	Ja	title
Kommentar	Ja	Ja	comment
Name	Ja	Ja	name
Datum und Uhrzeit	Nein	Ja	dateTimeString

entries.csv Struktur

- Pro Eintrag ein CSV Feld in folgender Reihenfolge:

<title, comment, name, dateTimeString>

Accountinformationen

- Bestandteile eines assoziativen Arrays (\$user), welches einen Benutzer repräsentiert
- Bestandteile eines Benutzers in users.csv
- HTML Formular-namen

Deutsch	In \$_POST von Anmeldung enthalten	In \$_POST von Registration enthalten	Ausliefern an Kursansicht	Variablenname / Input name
ID	Nein	Nein	Nein	id
E-Mail Adresse	Ja	Ja	Ja	email
Passwort	Ja	Ja	Nein	password

users.csv Struktur

- Pro Benutzer ein CSV Feld in folgender Reihenfolge

<id, email, password>

\$_SESSION Informationen

- Werden bei Anmeldung in \$_SESSION gespeichert

Deutsch	Quelle	Variablenname
ID	users.csv	id
E-Mail Adresse	\$_POST von login	email
Benutzername (nur für Admin)	\$_POST von admin login	username
Menge von erstellten Kursen (Kurse sind Arrays, die Kurseinformationen enthalten), wobei createdCourses[20] ein Kurs mit ID 20 ist	courses.csv	createdCourses
Menge von gekauften Kursen (Kurse sind Arrays, die Kurseinformationen enthalten), wobei boughtCourses[20] ein Kurs mit ID 20 ist	courses.csv	boughtCourses
Ist Admin? (nur für Admin)	Kam Anmeldung von admin login?	isAdmin

Lesen und interpretieren einer CSV Datei

- Es existieren für entries.csv, users.csv und courses.csv jeweils Key-Arrays, die aus einem Index einen Variablennamen machen

```
$entry_keys = {
```

```
    [0] = „title“,
    [1] = „name“,
    [2] = „dateTimeString“,
    [3] = „comment“
```

```
}
```

```
entries.csv:
```

```
    „Unverschämtheit“, „Basti92“, „12. November 2019“, „Ich will mein Geld zurück!!!“
```

```
while ($csvFeld = CSV Feld aus entries.csv) { //für jedes CSV Feld in entries.csv
    $entry = array(); //erstelle ein array, das ein gästebucheintrag repräsentiert
    $index = 0;

    //für jedes Element des CSV Feldes (zum beispiel „Unverschämtheit“)
    foreach ( $csvWert in $csvFeld)
    {
        // $entry[ 0 → $entry_keys → „title“ ] = „Unverschämtheit“
        $entry[$entry_keys[$index]] = $csvWert;
        $index++;
    }
}
```

- Die Key-Arrays sind unten in Globale Variablen definiert

Wieder-einlesbares Schreiben in eine CSV Datei

- Um sicherzustellen, dass ein assoziatives Array in einer vorhersehbaren Reihenfolge in die Zielfeile geschrieben wird, muss ein neues Array „toWrite“ aus dem input-array erstellt werden, welches nach der Reihenfolge der key-arrays bestückt wird

```
// $array soll in courses.csv geschrieben werden
function writeInfile($array)
{
    $toWrite = array(); //neues Array
    // Beispiel für $key: 0, Beispiel für $value: „title“
    foreach ($key, $value in $course_keys)
    {
        //toWrite[0] = $array[0 → $course_keys → „title“]
        toWrite[$key] = $array[$course_keys[$key]];
    }
    fputcsv($toWrite);
}
```

Verzeichnisstruktur

/

bilder

joud.jpg

course-images

0.jpg

1.jpg

2.jpg

instructor-images

0.jpg

1.jpg

2.jpg

css

style.css

js

script.js

lib

primaerfunktionen.php

sekundaerfunktionen.php

hilfsfunktionen.php

ausnahmen.php

private

data

users.csv

entries.csv

courses.csv

do.php

anmelden.php

registrieren.php

about_us.php

shop.php

gaestebuch.php

kurs_erstellen.php

kurs_bearbeiten.php

erstellte_kurse.php

gekaufte_kurse.php

kurs.php

Globale Variablen

- Müssen händisch eingetragen werden

Variablenname	Bedeutung
\$COURSE_KEYS	Key-Array für Kurse
\$ENTRY_KEYS	Key-Array für Gästebucheinträge
\$USER_KEYS	Key-Array für Benutzer
\$COURSES_PATH	Pfad zu courses.csv
\$USERS_PATH	Pfad zu users.csv
\$ENTRIES_PATH	Pfad zu entries.csv
\$COURSE_IMAGE_PATH_ABS	Absoluter Pfad zu den kurs bildern für interne repräsentationen
\$INSTRUCTOR_IMAGE_PATH_ABS	Absoluter Pfad zu den kursleiter bildern für interne repräsentationen
\$COURSE_IMAGE_PATH	Pfad zu den kurs bildern für interne repräsentationen
\$INSTRUCTOR_IMAGE_PATH	Pfad zu den kursleiter bildern für interne repräsentationen

```
$COURSE_KEYS = array(  
    0 => „id“,  
    1 => „title“,  
    2 => „summary“,  
    3 => „courseImagePath“,  
    4 => „priceCent“,  
    5 => „priceEuro“,  
    6 => „educationalContent“,  
    7 => „suitableFor“,  
    8 => „materialContent“,  
    9 => „instructorName“,  
    10 => „instructorContact“,  
    11 => „instructorImagePath“,  
    12 => „hImpairedSuitability“,  
    13 => „vImpairedSuitability“,  
    14 => „courseImageAlt“,  
    15 => „creatorId“,  
);  
  
$ENTRY_KEYS = array(  
    0 => „title“,  
    1 => „comment“,  
    2 => „name“,  
    3 => „dateTimeString“,  
);
```

```
$USER_KEYS = array(  
    0 => „id“,  
    1 => „email“,  
    2 => „password“,  
);  
  
$COURSES_PATH = „/private/data/courses.csv“  
$ENTRIES_PATH = „/private/data/entries.csv“  
$USERS_PATH = „/private/data/users.csv“  
  
$COURSE_IMAGE_PATH_ABS = $_SERVER["DOCUMENT_ROOT"] . „/bilder/course-images“  
$INSTRUCTOR_IMAGE_PATH_ABS = $_SERVER["DOCUMENT_ROOT"] . „/bilder/instructor-images“  
  
$COURSE_IMAGE_PATH = „/bilder/course-images“  
$INSTRUCTOR_IMAGE_PATH = „/bilder/instructor-images“
```

Exceptions

- Ausgeworfen von Sekundärfunktionen und Primärfunktionen
- Definiert in ausnahmen.php

Deutsch	Name
User wurde nicht gefunden	userDoesntExistException
Kurs wurde nicht gefunden	courseDoesntExistException
Anmelden ist fehlgeschlagen	loginException
Logindaten sind nicht vollständig	loginFieldEmptyException
Registrierungsdaten sind nicht vollständig	registerFieldEmptyException
Kursdaten sind nicht vollständig	courseFieldEmptyException
Benutzerdaten sind nicht vollständig	userFieldEmptyException
Gästebucheintrag daten nicht vollständig	entryFieldEmptyException
Kurs wurde bereits gekauft	alreadyBoughtException
(Benutzer-) Session existiert nicht	notLoggedInException
Kurs wurde nicht vom Benutzer erstellt	courseNotOwnedException
(Benutzer-) Session existiert bereits	alreadyLoggedInException
Benutzer ist bereits registriert	alreadyRegisteredException
Hochladen einer Datei fehlgeschlagen	fileUploadFailedException

Hilfsfunktionen

- Benutzt von Primärfunktionen und Sekundärfunktionen
- Datenbestandsfunktionen, CSV Datei Schnittstelle
- Einfache Suchfunktionen

Deutsch	Funktionsnamen	Aufrufende Primärfunktion	Aufrufende Sekundärfunktion	Parameter	Rückgabe	(kaskadiert) Geworfene Ausnahmen
Zählen von Kursen in courses.csv	countCourses	addCourse		-	Anzahl der CSV Felder / 2	
Zählen von Benutzern in users.csv	countUsers	register			Anzahl der CSV Felder	
Komplettes Lesen und Interpretieren der users.csv	readUsersFile	login		-	Zweidimensionales Array bestehend aus Benutzern wobei indizes nicht user ids sind	
Komplettes Lesen und Interpretieren der courses.csv	readCoursesFile	login, deleteCourse, editCourse, buyCourse	(ziemlich viele, siehe „Effekte von Sekundärfunktionen“)	-	Zweidimensionales Array bestehend aus Kursen, wobei indizes nicht kurs ids sind	
Komplettes Lesen und Interpretieren der entries.csv	readEntriesFile		generateEntries	-	Zweidimensionales Array bestehend aus Gästebucheinträgen	
Komplettes Schreiben der courses.csv	writeCoursesFile	deleteCourse, editCourse, buyCourse		courses (Array bestehend aus Kursen)	-	courseFieldEmptyException

Schreiben (hinzufügen) eines einzigen Kurses in courses.csv	writeCourse	addCourse		course (Array bestehend aus Kursinformationen) participants (Array bestehend aus Benutzer Ids von Benutzern, die am Kurs teilnehmen)	-	courseFieldEmptyException
Schreiben (hinzufügen) eines einzigen Gästebucheintrags in entries.csv	writeEntry	addEntry		entry (Array bestehend aus Gästebucheintragsinformationen)		entryFieldEmptyException
Schreiben (hinzufügen) eines einzigen Benutzers in users.csv	writeUser	register		user (Array bestehend aus Benutzerinformationen)		userFieldEmptyException
Suchen in einem Array bestehend aus Kursen nach Kurs mit Kurs ID	findCourseById	deleteCourse, editCourse, buyCourse	generateParticipantsList	courses (Array bestehend aus Kursen), courseId	courseIndex (Index in \$courses an dem Id gefunden wurde)	courseDoesNotExistException
Suchen in einem Array bestehend aus Kursen nach Kursen, die	getBoughtCourses	login		courses (Array	courses (Array	

von einem Benutzer gekauft wurden				bestehend aus Kursen), userId (nach zu suchender userId)	bestehend aus Kursen, die benutzer mit userId gekauft hat, wobei die indizes in courses kurs Ids sind)	
Suchen in einem Array bestehend aus Kursen nach Kursen, die von einem Benutzer erstellt wurden	getCreatedCourses	login		courses (Array bestehend aus Kursen), userId (nach zu suchender userId)	courses (Array bestehend aus Kursen, die benutzer mit userId erstellt hat, wobei die indizes in courses kurs Ids sind)	
Suchen in einem Array bestehend aus Benutzern nach Benutzer mit Email	findUserByEmail	login, register		users (array bestehend aus Benutzern), email (nach zu suchender email)	user (Benutzer mit email email)	userDoesntExistException
Suchen in einem Array bestehend aus Benutzern nach Benutzer mit ID	findUserId		generateParticipantsList	users (array bestehend aus Benutzern), userId (nach zu suchender ID)	user (Benutzer mit id userId)	userDoesntExistException

Primärfunktionen

- Aufgerufen von do.php
- Rufen Hilfsfunktionen auf

Deutsch	Funktionsnamen	Ausführer	Parameter	Rückgabetyt	(kaskadiert) Geworfene Ausnahmen
Anmelden	login	Benutzer ohne Session oder mit Admin Session	email, password	-	userDoesntExistException, loginException, alreadyLoggedInException, loginFieldEmptyException
Registrieren	register	Benutzer ohne Session oder mit Admin Session	email, password	-	alreadyRegisteredException, alraedyLoggedInException, userFieldEmptyException, , registerFieldEmptyException
Kurs löschen	deleteCourse	Benutzer mit Session, und isAdmin ist True	Id (des Kurses)	-	courseDoesntExistException, courseNotOwnedException, notLoggedInException
Kurs bearbeiten	editCourse	Benutzer mit	course	-	courseDoesnt

		Session, und Kurs ID ist in createdCourses	(Array mit Kursinformationen)		ExistException, notLoggedInException, courseFieldEmptyException, fileUploadFailedException, courseNotOwnedException
Kurs hinzufügen	addCourse	Benutzer mit Session	course (Array mit Kursinformationen)	-	notLoggedInException, fileUploadFailedException, courseFieldEmptyException
Kurs kaufen	buyCourse	Benutzer mit Session, wobei Kurs ID nicht in boughtCourses vorkommt	courseId	-	notLoggedInException, alreadyBoughtException
Gästebucheintrag verfassen	addEntry	Jeder	entry (Array mit Gästebuchentryinformationen)	-	entryFieldEmptyException

Effekte der Primärfunktionen

```
login(email, password):
    [Falls benutzer session nicht existiert]
    [Falls email oder password leer sind, throw new
loginFieldEmptyException]
    [$accounts = readUsersFile()]
    [$account = findUserByEmail(email)]
    [falls password übereinstimmt]
        [$SESSION[„id“] = $account[„id“]
        [$courses = readCoursesFile()]
        [$boughtCourses = getBoughtCourses($courses,
$SESSION[„id“])]
        [$SESSION[„boughtCourses“] = $boughtCourses]
        [$createdCourses = getCreatedCourses($courses,
$SESSION[„id“])]
        [$SESSION[„createdCourses“] = $createdCourses]
    [falls nicht, throw new loginException]
    [falls doch, throw new alreadyLoggedInException]

register(email, password):
    [falls benutzer session nicht existiert]
    [falls email oder password leer sind, throw new
registerFieldEmptyException]
    [try
    {
        [$account = findUserByEmail(email)]
    } catch (userDoesntExistException) {
        [$account = new array() ... [„email“] ... [„password“] ...]
        [$account[„id“] = countUsers()]
        [writeUser($account)]
        [login(email, password)]
        [return]
    }]
    [weil benutzer mit email bereits existiert, throw new
alreadyRegisteredException]
    [falls doch, throw new alreadyLoggedInException]

deleteCourse(courseID):
    [Falls benutzer session existiert]
    [$courses = readCoursesFile()]
    [$index = findCourseById($courses, courseID)]
    [falls $courses[$index][„creatorId“] gleich $SESSION[„id“]]:
        [unset($courses[$index])]
```

```

        [writeCoursesFile($courses)]
        [falls nicht, throw new courseNotOwnedException]
[Falls nicht, throw new notLoggedInException]

```

```

editCourse(course):
    [Falls benutzer session existiert]
        [$courses = readCoursesFile()]
        [$index = findCourseById($courses, $course[„id“])]
        [falls $courses[$index][„creatorId“] gleich $_SESSION[„id“]]
            [falls isset($course[„instructorImage“]):
                [$course[„instructorImage“] in
$INSTRUCTOR_IMAGE_PATH_ABS unter $course[„id“].jpg speichern, falls fehler throw
new fileUploadFailedException]
                [$courses[$index][„instructorImagePath“] = ...
                [falls isset($course[„courseImage“]):
                    [$course[„courseImage“] in $COURSE_IMAGE_PATH_ABS
unter $course[„id“].jpg speichern, falls fehler throw new
fileUploadFailedException]
                    [$courses[$index][„courseImagePath“] = ...
                [sonstige sachen in $courses[$index] manuell mit $course
sachen überschreiben. Falls dabei ein feld leer ist, throw new
courseFieldException).]

```

```

        [writeCoursesFile($courses)]
        [falls nicht, throw new courseNotOwnedException]
[falls nicht, throw new notLoggedInException]

```

```

addCourse(course):
    [Falls benutzer session existiert]
        [id = countCourses()]
        [$course[„id“] = id]
        [$course[„creatorId“] = $_SESSION[„id“]]
        [$course[„instructorImage“] in $INSTRUCTOR_IMAGE_PATH_ABS unter
$course[„id“].jpg speichern]
        [$course[„instructorImagePath“] = ...
        [$course[„courseImage“] in $COURSE_IMAGE_PATH_ABS unter
$course[„id“].jpg speichern]
        [$course[„courseImagePath“] = ...

        [unset($course[„courseImage“])]
        [unset($course[„instructorImage“])]

        [writeCourse($course, array())]
        [$SESSION[„createdCourses“][$course[„id“]] = $course]
[Falls nicht, throw new notLoggedInException]

```

```

    buyCourse(courseId):
        [Falls benutzer session existiert]:
            [Falls Kurs mit courseId nicht bereits in
$_SESSION[„boughtCourses“] ]:
                [$courses = readCoursesFile()]
                [$index = findCourseById($courses, $courseId)]
                [$courses[$index][„participants“] um $_SESSION[„id“]
erweitern]
                    [writeCoursesFile($courses)]
                    [$course = $courses[$index]]
                    [$_SESSION[„boughtCourses“][$course[„id“]] = $course]
            [Falls doch, throw new alreadyBoughtException]
        [Falls nicht, throw new notLoggedInException]

    addEntry(entry):
        [dateAndTime mit php funktionen herbekommen]
        [writeEntry($entry)]

```

Sekundärfunktionen

- Benutzt von öffentlichen PHP Dateien
- Sekundärfunktionen benutzen Hilfsfunktionen

Deutsch	Funktionsname	Aufrufende öffentliche PHP Datei	Parameter	Rückgabe
Weiterleiten an Seite, falls Benutzer nicht angemeldet ist.	redirectIfLoggedOut	gekaufte_kurse.php, erstellte_kurse.php, kurs_erstellen.php, kurs_bearbeiten.php	Link zur weiterzuleitenden Seite	-
Weiterleiten an Seite, falls Benutzer angemeldet ist	redirectIfLoggedIn	anmelden.php, registrieren.php, ,	Link zur weiterzuleitenden Seite	-
Generieren von HTML Code zur Anzeige von Kursobjekten in shop.php	generateShopCourses	shop.php	amount (Anzahl von Kurs Objekten), offset (Angabe ersten indexes, ab dem amount Objekte erstellt werden)	HTML Code
Generieren von HTML Code zur Anzeige von gekauften Kursen	generateBoughtCourses	gekaufte_kurse.php	-	HTML Code
Generieren von HTML Code zur Anzeige von erstellten Kursen	generateCreatedCourses	erstellte_kurse.php	-	HTML Code
Generieren von HTML Code zur	generateParticipantsList	kurs.php	kursId (ID des Kurses)	HTML Code

Anzeige einer Teilnehmerliste bestehend aus E-Mail Adressen				
Generieren von HTML Code zur Anzeige von Gästebucheinträgen	generateEntries	gaestebuch.php	amount (Anzahl von Kurs Objekten), offset (Angabe ersten Indexes, ab dem amount Objekte erstellt werden)	HTML Code

Effekte der Sekundärfunktionen

redirectIfLoggedOut(link):

[falls nicht isset(\$_SESSION[„id“]):
[zu \$link weiterleiten]

redirectIfLoggedIn(link):

[falls isset(\$_SESSION[„id“]):
[zu \$link weiterleiten]

generateShopCourses(amount, offset):

[\$courses = readCoursesFile()]

[\$subcourses = array_slice(\$courses, \$offset, \$amount)] //eventuell noch überprüfen ob

überhaupt so viele courses da sind

[\$outputHtml = „“]

[für jeden kurs in subcourses, hinzufügen von HTML an \$outputHtml]

dabei darauf achten, dass ein input element mit type=“hidden“, name=“id“ und value=“kurs id“
in den kauf knopf formularen versteckt wird

[ausgabe von \$outputhtml]

generateBoughtCourses():

[\$courses = readCoursesFile()]

[\$outputHTML = „“]

[für jeden \$course in \$_SESSION[„boughtCourses“]:

[generiere HTML aus \$course, hinzufügen von HTML an \$outputHtml]

dabei darauf achten, dass beim klicken nach kurs.php?id=\$course[„id“] verlinkt wird

[ausgabe von \$outputhtml]

generateCreatedCourses():

[\$courses = readCoursesFile()]

[\$outputHTML = „“]

[für jeden Kurs in \$_SESSION[„createdCourses“]:

[generiere HTML aus \$course, hinzufügen von HTML an \$outputHtml]

dabei darauf achten, dass beim klicken nach kurs.php?id=\$course[„id“] verlinkt wird

[ausgabe von \$outputhtml]

generateParticipantsList(kursId):

[\$course = \$_SESSION[„createdCourses“][kursId]]

[\$emailArray = array()]

[\$users = readUsersFile()]

[für jede \$userid in \$course[„participants“]:

[\$user = findUserById(\$users, \$userid)]

[\$emailArray um \$user[„email“] erweitern]

[\$outputHTML = „“]

[für jede \$email in \$emailArray]:

[\$outputHTML um generiertes HTML aus \$email erweitern]

[\$outputHTML ausgeben]

```
generateEntries(amount, offset):  
    [$entries = readEntriesFile()  
    [$subentries = array_slice($entries, $offset, $amount)] //eventuell noch überprüfen ob überhaupt  
so viele entries da sind  
    [$outputHtml = „“]  
    [für jeden entry in subentries, hinzufügen von HTML an $outputHtml]  
    [ausgabe von $outputhtml]
```

Primärfunktion auslösende HTML Formulare

- name attribut der formulare entspricht den namen der primärfunktionen (siehe bildbeschreibungen), action ist immer do.php

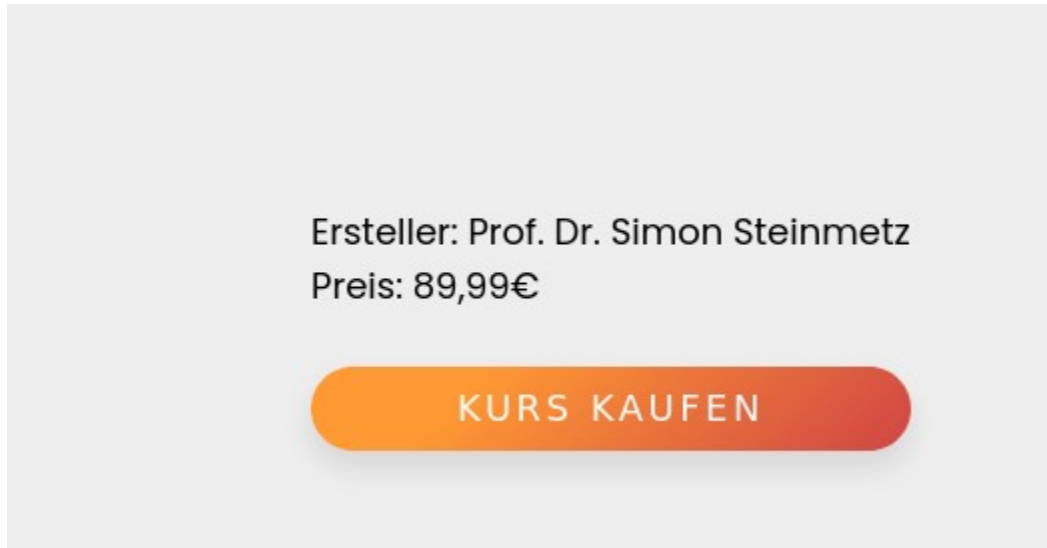


Figure 1: buyCourse aus shop.php

Implikationen:

- kurse in shopansicht müssen ein verstecktes input feld mit kurs ID haben

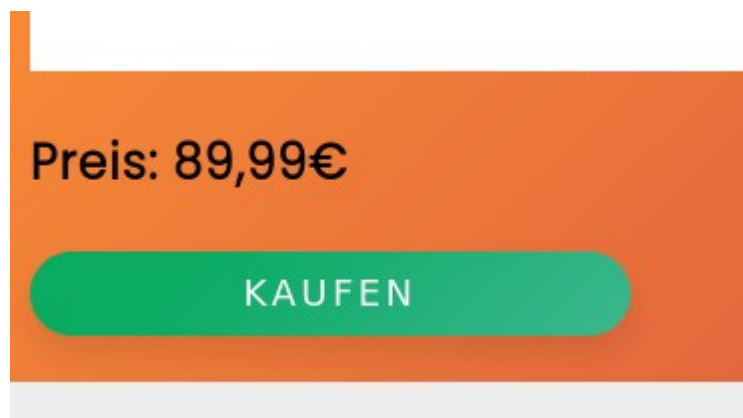
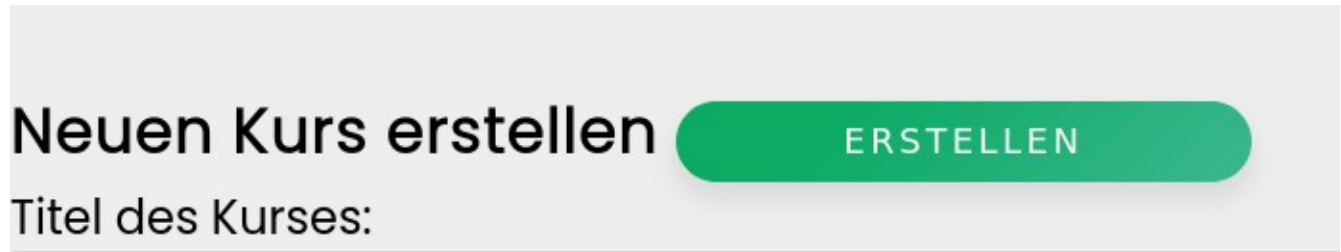


Figure 2: buyCourse aus kurs.php

Implikationen:

- kursansicht für nichtteilnehmer muss ein verstecktes input feld mit kurs ID haben



The screenshot shows a form titled 'Neuen Kurs erstellen'. To the right of the title is a green button labeled 'ERSTELLEN'. Below the title is a label 'Titel des Kurses:' followed by a text input field.

Figure 3: `addCourse` aus `kurs_erstellen.php`



The screenshot shows a form titled 'Kursbearbeitung'. To the right of the title are two green buttons: 'ÄNDERUNGEN VERWERFEN' and 'SPEICHERN'. Below the title is a label 'Titel des Kurses:' followed by a text input field.

Figure 4: `editCourse` aus `kurs_bearbeiten.php`

Implikationen:

- verstecktes input element mit kurs id



The screenshot shows a form with three input fields: 'Titel', 'Kommentar', and 'Name'. Below the 'Name' field is a button labeled 'Senden'.

Figure 5: `addEntry` aus `gaestebuch.php`

Registrieren

E-Mail Adresse:

Passwort:

Passwort wiederholen:

Registrieren

Figure 6: register aus registriere.php

Passwort:

Anmelden

Figure 7: login aus anmelden.php