

Projekt 2:

Rental Car App

Python Gen 5

In 3,5 Tagen entwickeln die Teilnehmer in 2-er Gruppen eine App zur Buchung von Fahrzeugen aus einer Fahrzeugflotte.

Um dies erfolgreich umzusetzen muss ein Frontend mit HTML, CSS und JavaScript auf der einen Seite und ein Backend mit Python (Flask API) geschaffen werden, mit der auf eine SQL-Datenbank zugegriffen wird (sqlite3 oder sqlalchemy).

Abschließend wird die Applikation mit Azure veröffentlicht.

Die Bearbeitung des Projektes startet am 3.6.24 um 8:30 Uhr und endet am 7.6.24. um 12:30 Uhr. Die Präsentationen finden in zwei Räumen statt.

Präsentationstermine

Raum 1:

Start	Ende	Vorname	Nachname	Gruppe
12:30	12:50			
12:55	13:15			
13:20	13:40			
13:45	14:05			
14:10	14:30			
14:35	14:55			
15:00	15:20			
15:25	15:45			
15:50	16:10			

Raum 2:

Start	Ende	Vorname	Nachname	Gruppe
12:30	12:50			
12:55	13:15			
13:20	13:40			
13:45	14:05			
14:10	14:30			
14:35	14:55			
15:00	15:20			
15:25	15:45			
15:50	16:10			

Aufgabenstellung: Rental Car App

Stellt euch vor, ihr seid ein Entwicklungsteams, das eine innovative Cloud Web App für die Anmietung von Fahrzeugen konzipieren und implementieren sollt. Das Ziel ist es, eine benutzerfreundliche, effiziente und skalierbare Lösung zu schaffen, die es Kunden ermöglicht, Fahrzeuge einfach und unkompliziert zu buchen.

Du wirst dich mit verschiedenen Aspekten der Softwareentwicklung auseinandersetzen, darunter die Integration von Backend- und Frontend-Technologien und Datenbankmanagement. Der abschließende Teil deiner Aufgabe ist es auch, die App erfolgreich auf der Cloud-Plattform Azure zu veröffentlichen, um eine hohe Verfügbarkeit und Zuverlässigkeit zu gewährleisten.

Die zur Verfügung stehenden Fahrzeuge, deren Spezifikationen und Buchungen, sollen in einer Datenbank hinterlegt werden.

Folgende Features sollen in ihrer Webapp umgesetzt sein:

- Buchung eines freien Fahrzeugs auf der Website.

Implementieren Sie mindestens 2 weitere Features. Hier sind einige Ideen, aus denen sie sich bedienen können, oder überlegt euch eigene:

- Darstellungen eines Kalenders zur Vereinfachung der Buchung.
- Filterung bzw. Sortierung der verfügbaren Fahrzeuge nach ausgewählten Kriterien.
- Administrationsseite, in der die Buchungen bzw. Verfügbarkeiten der Fahrzeuge eingesehen werden können.
- Bestätigung der Buchungsanfragen z.B. durch eine Mail.
- Darstellung von Fahrzeugstandorten auf einer Landkarte.
- Erstellung eines einfachen Rechnungsdokumentes.
- Nutzung von Rabattcodes.
- ...

Natürlich müssen Sie kein Abrechnungssystem (z.B. Kreditkartenbuchung) implementieren.

Nutzt Git zur Versionierung ihres Codes.

Hinweis zum Deployment auf Azure:

Versucht abschließend, eure Webapp auf Azure zu veröffentlichen. Auch wenn das der letzte Schritt ist, bemüht euch nicht zu spät darum, da es hier immer zu unerwarteten Schwierigkeiten kommen kann. Wenn ihr feststellt, dass eine Veröffentlichung auf Azure im Zeitraum der Projektwoche nicht oder nur unter großem zeitlichen Aufwand möglich ist, halten sie Rücksprache mit den Dozenten, sodass diese sie dann von dieser Anforderung befreien. Kurz: Versucht das Deployment auf Azure, aber beißt euch nicht die Zähne daran aus.

Hinweis zur Präsentation:

Nutzt die 20 Minuten Präsentationszeit einerseits den Zuhörern alle wichtigen Features ihrer WebApp zu präsentieren. Zeigt, was euch selbst an eurer App besonders gefällt.

Andererseits stellt bei der technischen Darstellung der Webapp interessante Aspekte eurer Lösung vor, bei denen die Zuhörer etwas lernen könnten. Ihr müsst nicht alles oberflächlich erzählen, sondern dürft auch einzelne Themen im Detail besprechen.

Kurz: Geht davon aus, dass jeder weiß, was euer Projektziel war und gestaltet die Präsentation lebendig. Plant Zeit für Rückfragen ein.

Hinweis zur Codeabgabe:

- Zippt euer git-repository (das heißt nur den .git-Ordner).
- Benennt die .zip-Datei nach dem folgenden Schema:
PythonG5_Projekt_2_<Vorname1>_<Nachname1>_<Vorname2>_<Nachname2>.zip
- Ladet die zip hier hoch: <https://cloud.co-dex.de/s/eeEK8GmmJZj43DL>