

# LU06.A07 - Multiuser Todo-Liste



Anpassen der Model-Klassen, DAO-Klassen und Endpunkte, um eine Multiuser Todo-Liste zu implementieren.

## Vorgehen

### 0. Vorbereitungen

- Akzeptiere das GitHub Classroom Assignment
- Klonen dein persönliches Repository in die Entwicklungsumgebung

### 1. Anpassung der Model-Klassen

- Fügen Sie ein Attribut für die Benutzer-ID in der TodoItem-Klasse hinzu.

### 2. Anpassung der DAO-Klassen

- Modifizieren Sie die Methoden in den DAO-Klassen, um die Benutzer-ID bei CRUD-Operationen zu berücksichtigen.

### 3. Anpassung der Endpunkte

- Ändern Sie die Implementierung der Endpunkte in der Blueprint-Klasse, um nur die Todo-Elemente des angemeldeten Benutzers zu verarbeiten und zurückzugeben.
- Verwenden Sie die `current_user`-Variable von Flask-Login, um den angemeldeten Benutzer in den Endpunkten zu ermitteln.

### 4. Teste deine API

Via API-Testing.http-File im Workspace

oder via: [▶ Run in Postman](#)

### 5. Abgabe

Die Abgabe der Lösung erfolgt als Push in das persönliche GitHub-Repository.

⇒ GitHub Repo für externe Besucher

GitHub Repository <https://github.com/templates-python/m323-lu06-a07-multiuser>

Lernende am BZZ müssen den Link zum GitHub Classroom Assignment verwenden



© Kevin Maurizi

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/modul/m323/learningunits/lu06/aufgaben/multiuser>

Last update: **2024/03/28 14:07**

