💡 Course Project – Simple TODO App

📝 Project Summary:

As mentioned in the title, the requested project is a simple **TODO App** but we will make it a little bit tricky with more functionalities and more features, let’s have a look at the core features ;)

⭐ Core Features:

– User can create an account with name, email, and password.

– User can login to see his/her TODOs and the date every TODO was created on.

– User can add new TODO to his/her existing TODOs.

– User can update an existed TODO.

– User can delete any TODO.

🌟 Bonus Features:

– User can see his/her TODOs grouped by the day or the month TODOs were created on.

– User can make a group of TODOs so that he/she can track the finished and unfinished related tasks easily.

- User should receive a notification to remind him/her before TODO finish date by 30 min.

👨‍💻 Technology/Stack:

You are totally free to use between three main technologies **Nodejs(Javascript/TypeScript), .Net Core (C#)**, or **Django/Flask (Python)** to build your backend server **but you must use MongoDB as your database and Redis as a caching service**.

✅ Acceptance Criteria:

1. Implementing all **the core features** as **an API** or **a front end application and consuming the developed API on it.**

2. **Making sure** that all the **functionalities are working fine** as mentioned in the requirements down below.

🛎 Requirements (Technical):

- **Implementing authentication feature** which users can use to **signup** as a new user or **login** as an existing user.

- **Implementing CRUD operations** with the requirements:
  1. User can see his/her TODOs **with the date each TODO was created on and the date he/she expects to finish it.**

  2. User can delete any TODO finished or not finished.

3. User can modify each TODO and **the date of the modified TODO should change to be the date the user modified the TODO on.**

4. User should add new TODO to the list.

For the Bonus Features:

1. **(EASY)** Implementing **Read operation** which user can use to get his/her TODOs for **a specific passed month or day.** (You can just decide which the user chooses day or month by another attribute will be passed with the API request)

2. **(Medium)** **User can create** a **group** and then add any new TODO to this group. (If you will work on this feature try to implement it at the beginning).

3. **(Hard)** **Implementing a scheduled events system.** System should notify the user before the end date of each TODO (the way is up to you you can send a mail to user mail or on the browser) that the end time will be in 30 mins.

The hard part is done, let’s see what will you get if you implement this project.

⬇️

🎉 Evaluation and Prize

We will follow a specific criteria to evaluate this project with a simple points system!

Features	Weights
Creating the project as an API and implementing the core feature only.	**50** Points.
Creating the project as an API, implementing the core feature, and implementing a UI for the app.	**100** Points.
Implementing any of the bonus features.	+ **50** Points to your score for each feature (+ **150** for all).
Hosting the app on any free hosting servers.	+ **50** Points to your score.
Using git, uploading the project on Github, and using micro commits.	+ **50** Points to your score.

📅 **Project due date:** **One month**, counts from the last day of the course!

🏆 Prize

The winner will get one of the **Udemy** courses for free, the choice is totally up to the winner, choose and I will send it to you!

or

You get **300 EGP** transferred by Vodafone cash!

Waiting for your awesome work!

Thanks!