For the DOTNET API

The users can be part of three categories : admin, creator or consumer. The admin can modify, delete and create articles/reviews that are his or not, the creator can create modify or delete articles/reviews that are his and the consumet can only read/watch articles. The auth should be done with bearer and JWT.

The users have a profile and a preference. The profile and preference are about things like age or darktheme preference… and the models are already done.

The admin and creator users can create new articles that have a title and a piece of text. In the model these two are there but I would want a timestamp as well… so in the article feed the articles can be ordered by those timestamps.

All users can comment on the articles, so the comment table has a compound PK formed with the article_id and the user_id keys, and the comment text. This is already in the model.

The articles have many tags, and the tags table will be inserted into the tb on database update, as the user_categories already are. I will give you the list of all the tags to put there but for now if you want you can put 2, 3 mockup tags there. The relationship between the articles and article_tags is many to many, ant these are done in the model.

The article also has a list of adds, but this part of the model is not done yet, and I am not sure I want it yet so this is something to do in the end of the project.

The project should be done using repository pattern, and there is a IUserRepo.cs for test that I have done but you can redo it if you can. I want an Interface for each model, with getAll getById create update delete methods, an Interface for the login and register and for other features that might be needed along the line so that the repository is granulated and clear.

The controllers should work directly on the implemented interfaces of the repository pattern, so no DTO's. The preferred db id Microsoft Sql Server Express. I have used it already and I have the connection string in the appsettings.json so you need to change it there. If you want you can use other db but tell me so that you can walkthrough me so I know how to connect and setup myself :)).

If you have suggestions or stuff that are unclear don't hesitate to tell me.

For the Frontend:

The Angular frontend should have:

- a dashboard page with all the posted articles in chronological order
- a user profile page so the user can modify their profile and preferences settings
- the article page in which the users can write comments
- the creation page in which the admin or creators can write new articles. This should be a simple page with one input field for the title of the article and one textbox for the content of the article/review. Also, there should be a section in which the creator can select the tags for the article from the available ones.
- the user deletion or management page in which the admin can delete users or modify their

settings

The creators should have a button somewhere on the navbar or idk so they can switch from the user mode to cretor mode so they don’t modify something by accident.

All the design should be as mainstream as it could be… boxes taken from material ui or other framework with already implemented cards and navbars and so on… I don’t want anything custom… just something easy to do for you.