

Instructions

.NET Project

1. Create an **EditProfile** handler to update the profile. We only want to validate against the **DisplayName** here as we do not ask the user for a **Bio** when they register so we want to allow this to be optional.
2. Update the **ProfilesController** and add an endpoint for editing the profile. Hint: Don't forget to restart the API server!
3. Test the endpoint in Postman using the 3 requests in the Module 18 folder.
 - a. If we try and update the profile again without changing it we will get a 400 bad request as there were no changes to save to the Database.
 - b. Is there a way we can accept the same data coming in and tell EF the entity has been modified even if it has not?

Client Project

1. Create a Zod schema called editProfileSchema.ts so we can validate the 2 fields the user can edit. **DisplayName** should be required and **Bio** is optional.
2. Add a new mutation function in the useProfile hook. Ensure that the DisplayName property is updated in both the ['profile'] and ['user'] queries. Choose whichever approach you prefer as invalidating, setting on success or optimistic updating are all viable options
3. Create a **ProfileEditForm** component in the Profiles feature folder and ensure this can be opened and cancelled in it's parent component, the **ProfileAbout.tsx**. We need a normal text input and a multi-line text input for the 2 fields.
4. Test it works!

Feature complete!

You can now commit your changes. Suggested commit message: "End of section 18".