I am here to document the page Edit My Profile.

First of all, I agree that it should be accessible from your profile and not from the side navbar. However, I was very tight on time because we had to prepare a demo for our product owner, so it stayed there for now.

So – when you are logged in, you get this as an additional Nav Bar item.

A green screen with white text

Description automatically generated with low confidenceWe all know how that works.

And this is how it essentially looks:

Graphical user interface, application

Description automatically generated

If you wanna change your first name, you put just that. If you wanna change your password, you have to input the old one and then the new one two times. And if you wanna change your address, you have to input all the fields again.

To implement this, I had to make a new method all the way from Blazor to the Database.

This was actually fine, because with the repository methods, it was very easy to edit an existing User.

I started by making UpdateDtos for these three types of updates:

* Change first name
* Change password
* Change address

I know that it should also be possible to change the phone number, but this part of the Domain had not yet been implemented when I was making the editing, so that does not exist for now.

The Update Dtos have the fields requested above and the username of whoever is requesting to be edited. I did not make it possible to change your username because that would imply updating foreign keys, and I did not want to deal with the consequent complications of that. We are too far into the Project Period for that lol.

Here is an example of one of those Update Dtos:

Graphical user interface, application

Description automatically generated

Then it’s the usual. It goes from the HttpClient to the endpoint (which was a PAIN in the ass to reach because Blazor doesn’t always update Endpoints when it rebuilts so a typo turned into many hours of pain lol), the endpoint communicates with the Application layer, this one delegates to the DAO layer, in which we had to update the protofile to be able to deal with this type of request, and then C# communicates with Java through Grpc (so we also updated the protofile in Java duh), and in Java the Grpc Implementation class delegates to the UserService which gets the user at hand, changes the fields that we changed (and it knows depending on the method that is called) and saves it with the repository.

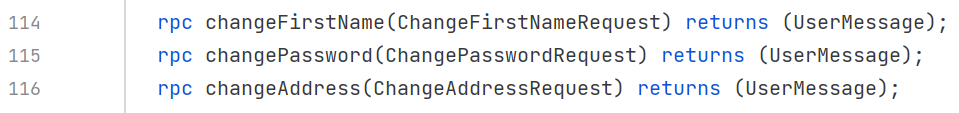
I guess the only thing there might be to show for this is the new Proto file changes:

New request types:

Table

Description automatically generated with medium confidence

New remote procedure calls:



And an example of how the User Service is dealing with the change:

Text

Description automatically generated

I think that’s all there could be to say about editing the profile. The page looks like shit but I was in a rush and we decided not to spend too much time on styling anymore, so I’m leaving it coz functionality-wise it’s okay. I should change the password fields to actual password fields coz for now I am leaving them visible lol…