

Appendix

Initial Client interview

Me: So we've talked a bit about the idea for my program, right. It's going to be a debate website - sort of forum style - , but could you maybe give me some more specific ideas , for the sort of criteria you would want? Start super basic at first please as well, to build the foundations.

Student: Yeah sure. So I was thinking of a program where you would go on the website and you would see a post and then you could click on it and see a basic description of the debate. Then you would see all the different arguments that are like for and against or whatever style of debate you choose.

Me: Ok that's good , thank you. Who would create the debates? Like a uni or the students? And would you need to have an account?

Student: I think having an account where users can create debates (and also the arguments too obviously) would be best. But like I don't want my password getting leaked too, can you make it secure?

Me: Yes, I can encrypt or hash the login details. Is that okay?

Student: Yes

Me: Perfect. So more about the debate, what sort of details should be in the debate and the arguments?

Student: So the debate should have a title statement or question , some short explanation and a topic. Then the argument should have like a title or thesis, a paragraph at least of their point / explanation of their thesis. They should also have the option to reference other people's arguments.

Me: Hmmm, that's interesting, that leads to my next question quite well. I was going to ask if users should be able to give weight to posts? And if that should be a sort of like dislike system or that should be based on the references.

Student: The references may be to disprove it, so it should not add to the weighting. I do think the like dislike system is quite nice though. Also, I was wondering if you could have it so that you could save specific topics, like if you found a topic you liked, like Nietzsche or something you could follow it?

Me: Yeah, anything you want I can add, and that sounds reasonable. Anything else like that?

Student: If you give me a second to think maybe?

Me: Okay, do you mind if I ask you some questions in the meantime instead then?

Student: Please, go on.

Me: So we have users that will make the debate posts, how should people view these posts? So obviously you have the home page, but then when you are specifically browsing - like - do you browse by category or by user?

Student: Not to be annoying but could I have both?

Me: Sure! Ok so more now about like accessibility right, so what are you going to use to access it? I know you don't have a phone right now, so I'm talking more about a browser, right?

Student: Yeah I use chrome mainly, sometimes brave. I want to be able to use it on both my laptop and PC.

Me: Okay, more about UX right. Use-er. Ex-per-ience. Okay, so you know you're using it, what sort of feel do you want?

Student: I want it to be sort of like the major social media sites. It just takes a few seconds to load a lot of posts in your feed. Also I want it to look pretty minimalistic and modern, nothing crazy; just less is more y'know?

Me: Ok perfect, that sounds like everything. Thank you.

Intermediary client interview

(This involves me streaming the backend API which I also do in section D)

Me: Hello, could we talk about the debate website?

Student: Sure, what's up?

Me: I am going to stream my screen, I have completed the backend of the product and I want to just make sure all the functionality is there. This might look a little weird though.

Student: Ok, I'm excited to see it. Oh that's not what I was expecting.

Me: Yeah, I haven't finished yet, so don't worry too much for now. But I wanted to make sure that all the functionality and security components were there before I do the front end and make everything look pretty, alright.

Student: Okay.

Me: So I have the checklist of everything you wanted here right, so could we just go through that. I have all this dummy data preloaded, and I'm just going to show you the functions and let me know if there are any needed before I actually like make the so called "fun part", because this is the actual important stuff.

Student: Yeah, yeah. Please show me.

Me: Okay, so we have the register function here, the inputs the user submits are username, email, password and confirm password.

Student: The password is not encrypted.

Me: It is on the database, if I pull that up... This is just what the user submits, see?

Student: Oh okay cool.

Me: We have a sign in that uses either email or username and a password.

Me: We have an auto sign in, basically using tokens so that you don't have to keep logging in, the user doesn't input anything for that.

Me: We have the paginated posts function which is super cool, it lets you choose the topics (if your signed in or if you're browsing a topic like you wanted).

Student: Yeah actually thats cool, and I like how its customizable. Could you have it so that you could choose how it sorts posts?

Me: How do you mean?

Student: Like on youtube you can choose: most popular, newest, oldest, so could you add that.

Me: Yes, actually, its super extensible, and thats exactly what I want to be hearing at this stage before I add the front end so thank you; let me note that down real quick.

Me: Create a debate, the user inputs the title, topic and short description; everything else is generated by the server.

Me: This is the rate post function, so it has direction, basically I am thinking of an up button and a down weight button? Post id will be abstracted, you won't have to see that dont worry. Everything cool so far?

Student: Yeah.

Me: Alright, btw you can look at the right hand side, that has the data responded from the server, Its graphql so I don't always include everything, but with this function create argument, you can see I return the whole argument. Is that in the correct format, any extra data you want?

Student: No, that looks fine.

Me: Ok, next function is rate argument , same thing as rate post. Umm, fetch post, you can look at a single post, like you wanted.

Me: Fetch argument, same thing as fetch post. That's an empty tab, and addTopic...

Oh, this is to follow a topic. Okay, and that's everything. Any notes, or other functions?

Student: Actually, do you have to login every time, or can you just click the remember me checkmark?

Me: I did that intuitively, but that should be part of the checklist, thank you.

Me: Okay, thank you so much for doing this, I should be seeing you with the final product soon.

Student: Okay, looking forward to it.

Final client interview:

Student: Hello, I have been trying the website out over the last few days and I have collected some thoughts.

Me: Perfect, before you start could you sort of order it, so its positives first, bugs, minor issues and then any big changes if I were to continue developing this?

Obviously, just speak as things come up, but if you could do that, that would be nice.

Student: No problem, let's start with what's good. I think everything I asked for is there, and it's super nice to use, very pretty. Lavender is not a colour that you usually see on websites for some reason and it's very pretty. Functionally its spot on.

Student: However, there is one bug, you can follow a topic, but you can't unfollow a topic, you can't click the button after you follow it and if you refresh the page then it just asks you to follow it again and the button doesn't unfollow it still.

Me: Okay, that's quite annoying, yeah that would be a minor improvement for the app.

Student: Yeah, and then when you scroll through the posts it's really quick, which is nice.

Me; Yeah, it's running locally, so that's why it's so fast, but you are right, we could probably fetch 50 posts a page without issue rather than just 20.

Student: Mhm, about pages though, it's really annoying and not what I quite wanted, I wanted it to be like twitter, where you can just scroll down and then click a button that loads more posts and you can just keep scrolling. Infinite scroll, I think is the name?

Me: Okay thats another sort of improvement / quick fix that I'd do if we were to keep developing it.

Student: Also, I notice when I create a debate or an argument that I have to reload the browser when I go back to the home page or post page, especially if I have left another tab open, I want either it to automatically load or for there be a button to let you know that there are more posts.

Me: Okay so programmatically speaking that's a lot harder of a feature, that's more of an extension, because that notification system would require webs sockets; atleast in my mind. Okay, continue please.

Student: I like how you can look at individual arguments, I thought about asking for an edit feature, but that would make referencing and the debate aspect kind of annoying.

Me: Yeah I had the same thought.

Student: But what I would mention though, is that the layout is kind of annoying. If you were to keep developing it , I would want a structured layout with for and against on two side and the rejoinders at the bottom, Or for the different for, against, rejoinder arguments to be separated.

Me: Yeah, that would be another big change, because that would involve changing the queries significantly to load the for, against and rejoinder posts separately. Also it would seem tricky to lay out. But that's certainly a good extension. Anything else?

Student: I feel like I'm being overly negative, it's quick to use, fun to look at. I like how you can structure the queries in a bunch of different ways on the home page, and I like how easy it is to navigate to a topic or user.

Me: Well thank you for your compliments and your constructive criticism. Thank you so much for doing this.

Student: No, thank you so much for actually building this.