Technical Support Specialist Questions

# Q1. Discord is our main home for our creators and where we do most of our support. What experience do you have with Discord as a non-user? (eg. Moderator, owner)

Unfortunately, none, and can't say I've used it that much as a user, too, but I have experience in administering/moderating forums (phpBB and vBulletin engines were most prominent), as well as chatrooms (IRC and Shoutbox). Will probably have a learning curve in Discord, but I should be able to adapt quickly, as I learn quickly, which can be [confirmed](https://www.simbiat.dev/about/resume/#gallery=7) by my past colleagues.

# Q2. How familiar are you with support and dev tools such as JIRA, Zendesk or other support related platforms?

My main experience would be with ServiceNow, which I used in Citi for incident (tickets) management, problem management, change control, inventory and even group administration. I have used JIRA for reporting (and following-up on) bugs in Smartly.io, too, since it was the main tool there. Used Zendesk only for articles editing.

# Q3. Currently, our users on Discord are Creators. They vary from absolute beginners to experienced developers and therefore their technical problems will be very different. How would you approach dealing with these differences?

As usual, unless I somehow already know of the level of experience of the user, I would try to predict it from the original question or problem description. For example, if the user is using certain terminology in the description, it can be an indication of their knowledge in the respective subject.

That said, I would still not blindly assume anything, and if I have any doubts, I will find a way to ask some question to properly establish their level. The way I do that will differ based on circumstances, though, because I, myself, can get a bit frustrated, when I am asked a question which clearly shows, that the supporter assumes to be a total newb, even when my initial message may indicate otherwise.

Depending on the established level of knowledge/experience of the user I will try to adjust the wording of my explanations or questions. "Dumb them down" if necessary, provide screenshots with some markers, if they could be clearer than words, in case the user is less experienced. If it's a more experienced user, then I would probably be more technical and skip some of the lengthy explanations, where they may not be needed.

Regardless of the level, though, I would also try to educate them if needed and possible, so that they do not only get their problem resolved, but learn something new, that will "stick" with them, if possible.

# Q4. As Technical Support, you will have to work with both users and developer team. This means you will have to act as the middle point between those two. What experience do you have with this relationship, if any?

In Smartly.io I was L3 support, so most cases I got were either bugs or behaviors that were borderline bugs from user perspective, so I had to translate user issues from user's language to something more technical, that would turn either into a bug report or a feature/improvement suggestion, that developers could act upon. Obviously had to simplify any technical difficulties we had on our side when talking to customers, so that we could manage their expectations, but at the same time assure, that we will solve whatever the problem is soon.

In Citi… Oh, man, don't get me started on that one. I had a role of "application manager", which in short made me the 1st point of contact for any remotely technical question involving the applications assigned to me. Besides tickets or bug reports, I had to handle some business analysis even and properly convert business requirements into technical ones (although usually I was not the only person on the task). I also had to do the reverse as well, that is explain in business terms how certain technical limitations may be affecting respective business processes.

# Q5. Our creators and community are important to us, so we always want to make sure we provide them with ongoing communication and interaction from us. How would you ensure that we provide this level of relationship to them?

Ideally, you should have a community manager who actually plans out those interactions, it may not be a good idea to put this on tech support, unless you mean communication during ongoing issues.

But if I am to abstract myself from the role segregation, as per my own experience – transparency is key. Especially if we are talking about issues. For example, keep a list of known issues: this will show company's awareness of those, can be used to "track" activity by external stakeholders (or users), and it can serve as L0 support or so-called "support deflection tool", because before reporting the same issue again, users can check the list of known issue, find that the issue is known and… Not come to support.

If the service is dependent on some online services – provide a page that shows status of servers/services in a clear manner. If possible – split by components if that makes sense. For example, you have a huge product, which consists of multiple modules, and it is possible that 1 module goes down for whatever reason, but others can still work normally. You can indicate that on a status page. Although, if you do have any kind of outage, you need to have ways to increase awareness of users about it: send mails, ping in Discord, if it may be affecting revenue – have a way to calculate exact amount of time the outage lasted, even if you will not be compensating for the outage.

Oh, and most importantly: if you do screw up at some point – own it. Be honest. "We made a mistake and certain client data may be lost/corrupted. Here is the explanation of what happened on our side and what steps we have taken (or will take) to ensure this does not happen again". Do not try to hide your mishaps, do not try to put all the blame on some other party (even if they are to blame). This is difficult, but from my personal experience leading a small community based around a product – it builds loyalty, which is extremely important.

In addition to that, you obviously need to find ways to keep the audience engaged, for example through periodic community events, possibly with some small prizes even. But that's where you would want a proper community manager, because it's best to plan these things ahead.

# Q6. As part of the support platforms, you will be expected to provide the best ways to ensure our users get top level support from YAHAHA. What are some of the ways you will look to do this?

Difficult to say without knowing what you already have, to be honest.

Firstly, transparency. Yes, again, same as in previous question. Without it you can't build trust. Obviously, there may be some things, that you would not want to disclose at certain points, but this is something that you need to strive towards.

Secondly, be curious. Do not approach question from users as something ephemeral. Try to understand their use-case and what led them to that question. Try to understand what their goal is. Help them not just get an answer but get them closer to that goal. It's difficult to explain, sometimes it may be just a gut feeling like "Perhaps user is not looking for the answer to this question, but rather to some other, bigger question". If you get this feeling – act upon it. Ask the client for more context (but do not push for it), and if they provide it – try to help them solve their "bigger" problem, if you can. Worst case scenario, you can't really help them (it may be something they need to do themselves), but you will show that you care, which will, again, build loyalty.

Thirdly, and this may seem counter-intuitive – build tools for support deflection, otherwise known as L0 support. If a user comes to you with a question, you should always think, if there is anything you can do to prevent that question from occurring again. Perhaps, you should add an article to knowledgebase or a question to FAQ section somewhere. Or a tooltip to respective control in the app. Or maybe it would make more sense to redesign the UI for the feature to make things more obvious. Why? Because people do not really want to talk to your tech support. It takes time, it can take effort (especially for introverts), and if they do – it will cost your own company, because supporter will be working on that case when he may be doing something else. This is an on-going process, though.

And the most important thing: [do not ghost](https://www.simbiat.dev/talks/threads/135) your clients. Like never. If needed, come up with "we will look into it, but can't promise anything" kind of response, but do **not** get *silent*. I repeat: **never** ghost.

# Q7. Support can mean you get some mean or even hostile responses. How do you deal with these particular cases?

It depends. If the frustration is caused by something on our side – definitely need to use empathy and finds ways to assure, that we are fixing the problem as fast as we can. Maybe even find a way to compensate for it, if possible and makes sense. And do not just say "I understand your frustration", if possible, try to find possible workarounds. As mentioned before – get user context, what is their goal, maybe this is an alternative. If you can't do anything, always be ready to offer more detailed explanation about what's going on (transparency, as mentioned before), and keep in touch with the user, to let them know of the progress, especially, when the issue is fixed.

If aggressiveness is not caused by the product… That's a tough one without an example. If you can ignore it, because it does not affect the issue description or your ability to investigate and fix the issue, - ignore it. I mean, feel free to scream, curse or even punch some pillows in real life, but do not do any of that in the chat. If user's aggressiveness prevents you from doing your job – it is fine to let them know about that in some polite manner. Maybe this could be a possible situation, when you can ghost a user, but only for some limited time, because you will still want to check up on them, when they come down.

This is much easier with chat or email support. On call… You will need nerves of steel, and still follow the same principals.

# Q8. Currently, our Community team is based in Helsinki. If you are not located in Helsinki are you open to relocation? If you are located in Helsinki, what type of work setup do you prefer? (Remote, Hybrid, Office)

I am in Helsinki (Koivusaari), so no need for relocation. As for work setup… To be honest, I got used to remote work, since I've been working remotely since pandemic started. And I find that tech support itself does not really require your presence in the office. That said, I am not against working in the office.

# Q9. Why does YAHAHA interest you?

It is a gamedev company. I mean, yes, you are not creating the games yourselves, you seem to aim at providing the tools to create the games, but it's close enough for me, since I do want to get into gamedev or get as close to it as possible. Maybe Yahaha will turn out to be the best tool for my gamedev needs. At the least, I do have a game concept, that may work on such a platform.

And I am also looking for a job since June 2022, so… There's that.

# Q10. What salary range are you hoping to receive for the position?

The most difficult question for me 😅 I was relocated to Helsinki for 4300 euro gross per month, and the price was based on my extensive experience in tech support (and not only). I would be ecstatic to get as much (let alone more) but considering my personal situation (currently getting money only from YTK) and depending on scope of the actual position – I am ready to discuss [much] lower salary.

# Q11. What is your potential start date availability for joining the company?

Today.