

## BBC London Assessment Centre – Technical Questions

Name: Conor Sheppard

Technical Questions:

1. We're looking for people with a real passion for collaboratively creating great software. Please give an example of a software component you have designed and written from Concept to deployment, outlining the steps you took. (1000 character limit)

I have designed an Android application which worked as a smartcard for public transport. This was not a real-world application but a simulation.

The systems was made up of several components including

- The user's smartphone, which was the smartcard client application
- A terminal, off of which the user would tap their phone in the same fashion as you would an oyster card
- The backend system, a RESTful web application written in PHP. I did not use a framework and the backend consisted of a series of PHP scripts that ran various tasks.
- The database. SQL, relational.

The aim of this application was to demonstrate the power and use of NFC (Near Field Communication) harnessed via HCE (Host Card Emulation). HCE is a tech stack that allows you to use your smart device as a smartcard.

I first built the client side system and got used to working with HCE as I was not previously familiar with it. I then used TDD to build up a series of RESTful endpoints in order to pass data back and forth over the system.

2. Using the example that you provided above, tell us about a significant decision you made to solve a technical challenge. Give details of technologies that you chose and why you chose them. (1000 character limit)

When I initially designed the system, I was doing a lot of computations in the client side of the application - in the smart device itself.

It got to the point where it was too much for any given smartphone to handle. I started notice my app displaying odd behaviours and on closer inspection and debugging I realised that the main thread of the application was running out of memory and skipping over various tasks in order to perform garbage collection and memory allocation.

I had two options, either make the application multi-threaded or redesign the system in order for most of the computations to take place in the backend.

I chose the latter and redesigned the system.

The result was the system was faster, easier to manage and much more scalable than before.

3. Using the example that you provided above, tell us about how you ensured your software was fit for purpose and of high quality. What did you learn and what would you do differently next time to do a better job? (1000 character limit)

I ensured it was fit for purpose through test driven design and looking at popular design patterns of similar systems that were proven to work.

Every time I incorporated a new module into my system I would test it outright, on its own and make sure it was fit to incorporate it into the system as a whole. I would then conduct tests to see its performance now as part of the system.

I learned several lessons from the example above. One was not to try to do too much all in one place and to recognise the power of compartmentalised design where the various modules of a system don't have a strong dependency on one another.

Next time I would try to plan out a more distributed system and harness the power of a system where all its calculations and workload are placed on the server rather than the client so that, as the system grows, the backend can be expanded and new users' activity be load balanced more efficiently.

4. Please tell us about the ideal position you are looking for and also confirm your key technical skills? Please use this opportunity to tell us about a position you most prefer so we can ensure you are matched into the right position should you be successful (500 character limit)

Right now my key technical skills are working with server-side code, my previous role was as a Java Server-side Developer and I enjoy also working with Node.js. I do, however, like visual design and plan on moving toward more front end coding over the next few months.

I am good with data structures and algorithms, OOAD and multi-threading/concurrency.

I would be best placed in a role working with either Java or Javascript, these are the two languages I prefer, although I do like to pick up new languages/frameworks every now and again.