# Lab 1 Questions

1. **Why do we validate data before sending it to the server at client­side, as opposed to just letting the server validate data before using it? What we get and what we lose by it?**

The main reason is to avoid unnecessary data traffic and latency between the client and the server. This can improve performance and user experience since the person using the application doesn’t have to wait for a server response before noticing and being able to correct invalid data. The downside is that it is not as secure since malicious users potentially can bypass the client-side validation and send their own messages to the server by studying how the client behaves and what requests it sends. Also, some data validation will still have to take place on the server since only the server, for example, can detect if there is another user registered with the same email address so we end up with double validation.

1. **Why isn’t it a good idea to use tables for layout purposes? What would be the replacement?**

There are several reasons. The biggest one is that a table is limited in that it essentially is a grid where all cells in all rows at least have the same width. This can be circumvented in some ways but often leads to bad markup and ugly hacks to make things look nice. Tables are designed as being a way to represent a chunk of data and are not semantically correct to use for layout.

The obvious replacement is CSS-based layout where <div> is the main building block. This gives the developer a better abstraction between looks and structure and allows the layout to be changed while the same content is being displayed.

1. **How do you think Single Page Applications can contribute to the future of the web? What is their advantages and disadvantages from usage and development point of views?**

I like the idea of Single Page Applications. The main advantage is that only one HTML page is downloaded which means that the user URL doesn’t change. Even though calls to a server can be made, the user can potentially use the entire web application after the first HTML file (with CSS files and scripts etc.) has been downloaded and rendered. I see a clear advantage on mobile where you might see only a small part of the application at any given moment and loading a new HTML might break you out of a fluid web experience.

A disadvantage for developers is that it might be easier to get an overview of the application if different parts are split up on different pages and it might be easier for multiple developers to work on the same application but it is always important to put the user first.