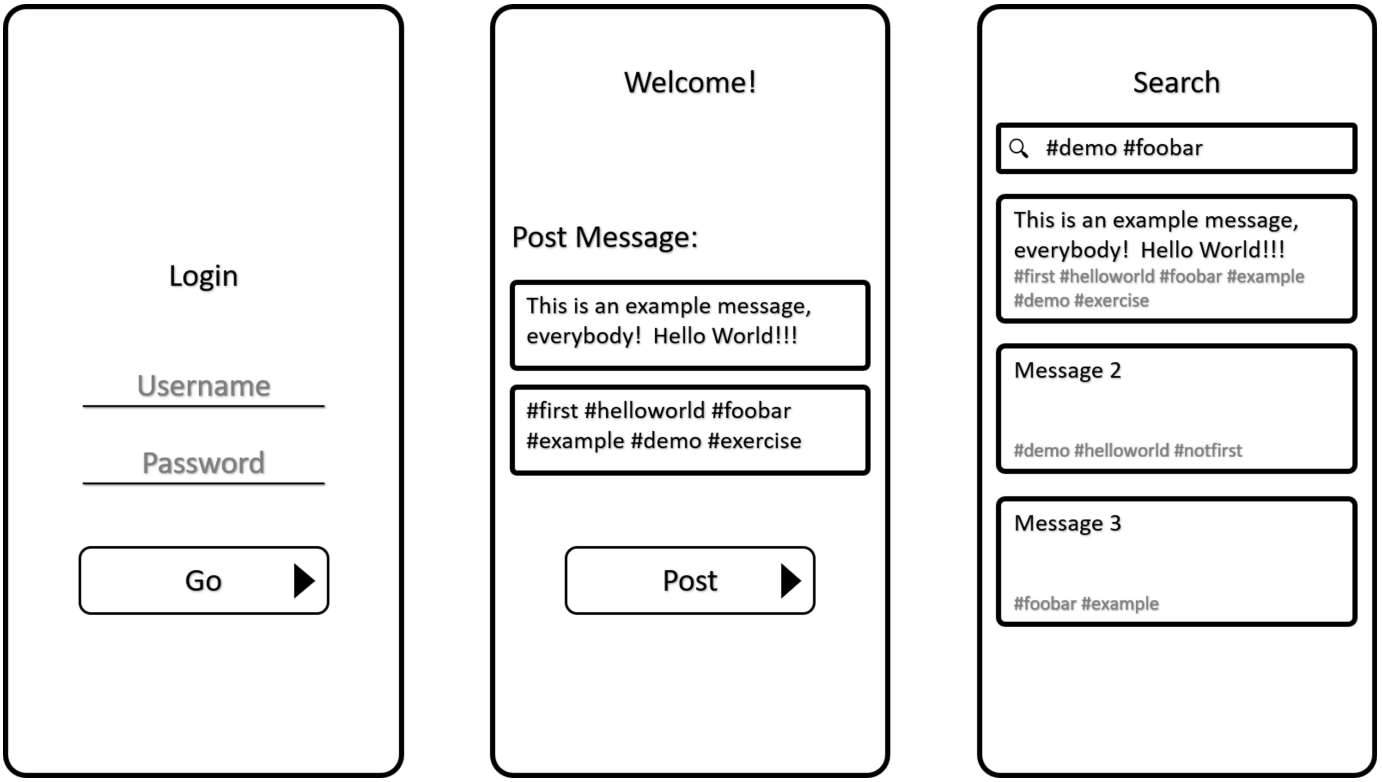
Backend Developer - Assessment

YINZCAM

This exercise focuses upon your ability to analyze a problem and determine a solution.
At YinzCam, a common requirement for developers is to design systems to meet certain needs, and it's an expectation that our developers be able to do this at a competent level.

For this exercise, we want you to design a system to accommodate the following application design. For the sake of simplicity, not all screens that you would expect in a fully fledged application are displayed.

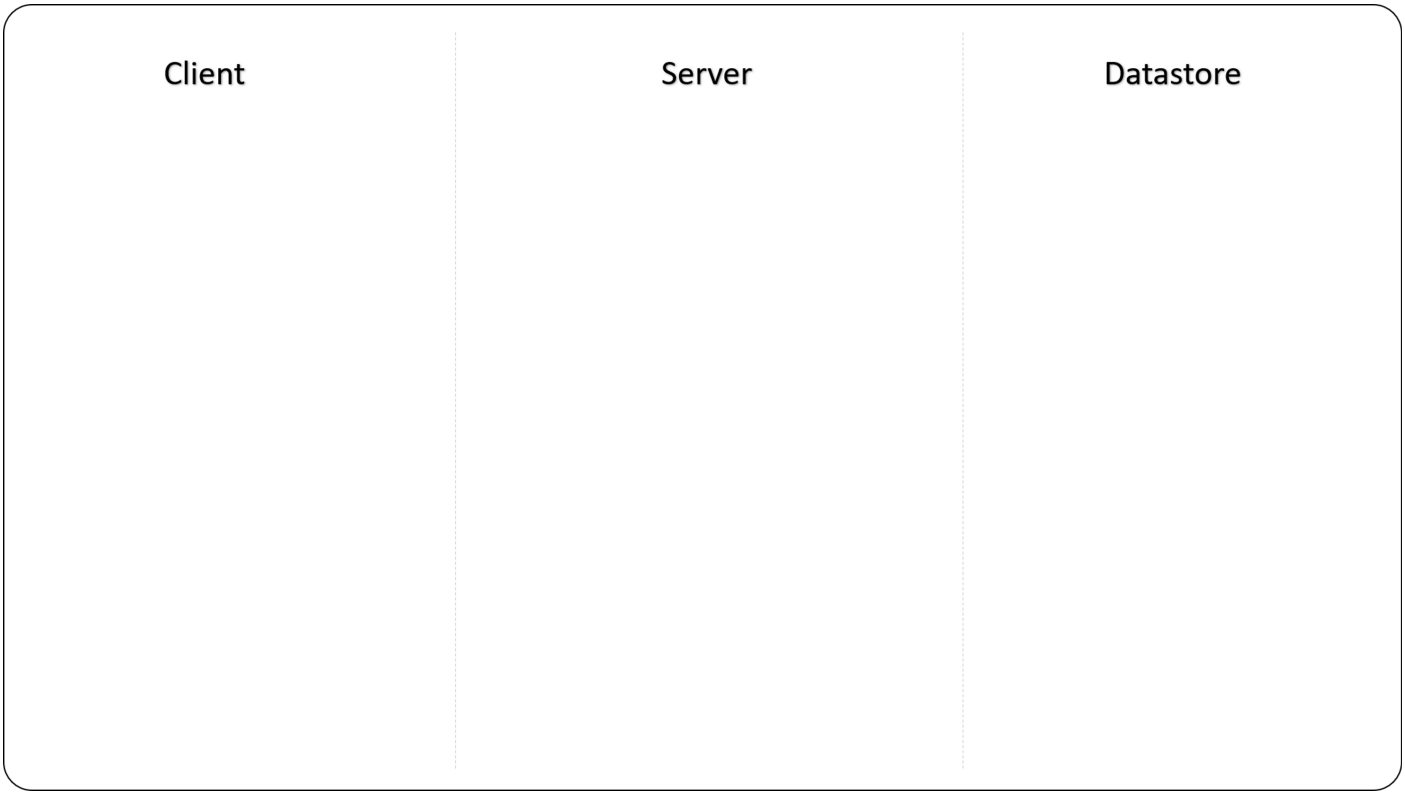


Here, we have a simple application where a user is prompted to login.
Once the user logs in, they have two options. They can either post new content or search for content.

For posting content, the user is able to type in a message along with a set of tags.
For searching content, the user can specify and set of arbitrary tags and messages that match those tags are displayed on the screen.

In a very simple sense, these are the product requirements. You are tasked with designing a system that can support such an application. Any information that is not directly specified, you may make a reasonable assumption, but be sure to note it and justify *why* you made it (why it was reasonable). You may do this exercise in whatever format is easiest for you (e.g., using an online tool, a text document, pen and paper, code, etc.).

As a starting point, this system probably has three high-level components. A client, server, and datastore.
It may be helpful to start filling in this diagram with all the various components you think may be necessary.
For example, servers, databases, networking devices, client devices, etc.



In addition to a high-level diagram showing the various components and how they interact, you should describe a few items in more detail.

- Communication method between client and server (e.g., protocol, how does login work, how do you post messages, how do you search for messages)
- Any code-level decisions you would make
- Your choice of datastore structure/technology
- Reasonings behind any design decisions you've made
- Any assumptions you've made

Overall, describe this system in as much detail as you feel necessary. As a good rule of thumb -- if you were to hand your document over to someone else, would they be able to both understand and implement your system? This problem is intentionally open-ended to see your thought process and understand your technical ability. If you have any questions, please don't hesitate to ask.