**Research for Login and Register System**

For the login and register pages there must be the programming language decided in the first place. After the programming language was chosen, tutorials were followed such as YouTube videos and Webpages:

**Sources:**

YouTube tutorials: [https://www.youtube.com/watch?v=0NFwF7L-YA8HYPERLINK "https://www.youtube.com/watch?v=0NFwF7L-YA8&t=701s"&](https://www.youtube.com/watch?v=0NFwF7L-YA8&t=701s)

<https://www.youtube.com/watch?v=tJVBXCNtUuk>

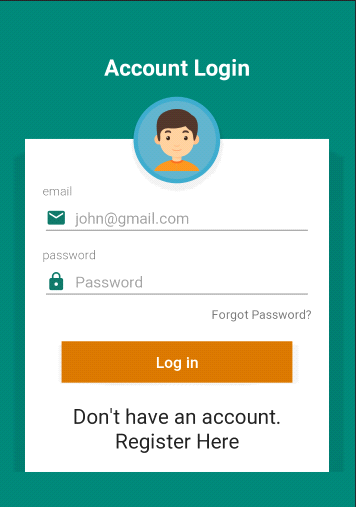
[https://www.youtube.com/watch?v=HRF8NpoFtegHYPERLINK "https://www.youtube.com/watch?v=HRF8NpoFteg&t=300s"&](https://www.youtube.com/watch?v=HRF8NpoFteg&t=300s)

Tutorials on websites: <https://stackoverflow.com/questions/42247465/how-to-connect-android-studio-with-sql-server-database>

<https://parallelcodes.com/connect-android-to-ms-sql-database-2/>

<http://seotoolzz.com/android/android-login-app-with-mssql-server.php>

**For the first template design, the result would look like this:**



**Overview of how the login system works**

When the user wants to login, he sends a login request, the server then encrypts a message of some sort (the challenge) and generates the correct response (IE encrypts the challenge using the stored hash). The challenge is sent to the client who calculates what the response should be (by taking the user's password and re-generating the hash from the password). The client sends this encrypted response to the server who compares it to the pre-calculated correct answer. If they match, the client must have the correct password in order to generate the correct response, and you let him have access.

The good part is that the password never gets sent to the server, except during the initial account creation stage, and even then, its sent as a hash.