



## Login Interface

We've got a backend for an accounts system, but we still need to set up an interface that allows users to register and log in. We can create such an interface from scratch, which is simple enough, but there's another approach for getting up and running even faster.

All we have to do is add another package to our project. This package is known as `accountsUI`, and it will create a fully functional interface for our user account system. To install this package, run the following command:

```
meteor add accounts-ui
```

Then, once the package is installed, switch to the HTML file and include a template named `loginButtons` between the body tags:

```
{{> loginButtons}}
```

## Testing the Interface

Save the file, return to the browser and notice that a "Sign in" button has appeared inside the interface. If we click this button, a login form will appear along with a link that says "Create account".

But this is not a dumb interface. Without any configuration, it's possible for users to register, log in and log out. To see this in action, create an account. You'll be logged in automatically after registration.

## Find and Fetch

Then use the `find` and `fetch` functions on the `Meteor.users` collection. You'll notice that a document is returned and this document contains the data for the account that we just created. At the moment though, there's no reason for a user to register or log in because signing up doesn't offer any kind of extra functionality.

To fix this, let's make it so unregistered users can't see parts of the interface that they're not supposed to see, like the list of players and the "Add Player" form. First, log out of the account. Then, return to the leaderboard template and create a conditional using the Spacebars syntax. This conditional should test whether or not `currentUser` returns `true`:

```
{{#if currentUser}}  
  {{> leaderboard}}  
{{/if}}
```

Because of this code, logged out users won't be able to see any of the content that should require users to be logged in. If you'd like to display a piece of content to logged out users in place of this form, you can add an `else` clause to the conditional.