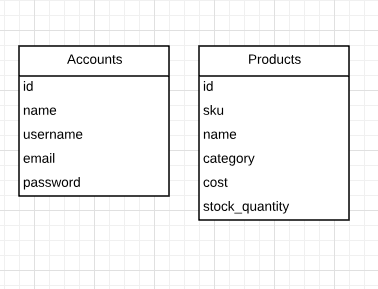
# Assignment 3

## Specifications:

* Login/Register Pages - When the site is navigated to, the login page will be the page that the user lands on. They will be prompted to login with their username and password and only once they have done so successfully will they be directed to the home page and granted access to the rest of the site. Alternatively, from here they can also access a create account page where they are able to create an account, filling in the necessary fields as requested. The validation is done with a mix of AJAX, XML and REGEX. We used AJAX to query whether the entered username is available, as well as to see whether the entered password and the validated password match. Regex was used to restrict entry such that only specified characters/ formats were able to be submitted. When all the fields are filled in as requested, the user may submit their information which will be saved in the database with an encrypted password.
* Search Page - Within our search page, we used AJAX to query the server asynchronously. We check if the text that the user has entered partially matches the products name, sku, or category and if it does, display the corresponding product amongst the rest of the products the search input also matches. This search is dynamic and will update the results with more specified results as the user continues to type.
* Top Menu - Across our whole website we have a navigation bar located at the top of the screen. This navigation bar includes a logo which is linked to the homepage, a home button, a browse page button, a search page button as well as a logout button. Each of these buttons will change colour when highlighted over and link to their corresponding pages.
* Products - Each product shows the following information when it is a result of a query: Stock Keeping Unit (SKU), Name, Cost, Category, Stock Quantity.
* Presentation - Our HTML is styled using a combination of our own CSS as well as with the use of BootStraps predefined styles.

## Database relationships

There are no relationships within our database. This is because we only required two tables, accounts and products, which weren’t related in any way.

## Instructions:

### Logging in:

When a user first visits the site, they will be presented with a login page. Here the user can log in if they already have an account. An account is supplied for you to log in with to test the site or you are free to create your own. The supplied account is:

Username: BobTheBuilder

Password: Bobthebuilder1

### Creating a new account:

When the user clicks on the "Create New Acccount" button, they will be brought to a page where the user can enter details needed to create a new account. This can also be navigated to via the navigation bar at the top of the site.

### Search Page:

When a user is logged in, they can click on the "Search" button in the top navigation bar and will be brought to the search page. Here, a user can search for products and will get near instant results.

## Design Choices:

### MVC

We used a Model, View, Controller design that allowed us to separate the necessary components. We kept all of our PHTML pages within their own directory as they only implemented view functionality, we kept all of our controllers within their own directory as they only control the flow of our website, and our model classes were also within their own directory as they allowed us to model our information that was stored on our server as well as perform actions using both the database and object information.

### CSS

For this assignment, we primarily used bootstrap's predefined tags to style our buttons, links, inputs, tables and menu amongst other elements. Bootstrap was also very useful for changing all the fonts to give the website a more modern look. We used CSS to get consistent styling across the whole site allowing for users to seamlessly navigate through a very familiar environment.

### PSR

We followed all PSR1 and PSR2 standards throughout our code and ran CodeSniffer ensuring all errors and warnings were righted.

### AJAX

AJAX was used in various places throughout our site. It is most noticeable in our live search where we asynchronously query the search the user has entered and in real-time update the HTML such that it updates the page with the results to the users search. It is also however used to validate information on the User Create page, and as such, that their desired username wasn’t already taken and that both their password and confirmed password match.