Medical Informatics (BIOM9450)

Tutorial 4: Client-Side Form Validation Using JavaScript

There were many functions and scripts used to complete this tutorial and, in this report, I would be going through the design choices for the JavaScript validation process.

The first part was to validate the email address, and this was done using the code below:

As shown from the code snippet, a regex expression is used to check for numbers and letters before and after the "@", and 2 to 4 characters after the "."

Next, there is the typing of the password and confirmation of password. The code snippet is as follows:

```
<!-- Adding Re Enter Password and Verifying It-->
     / class="form-group">
<label class="col-md-4 control-label" for="Password">Password</label>
              class="col-md-4">
           <input id="Password" name="Password" type="password" placeholder="Enter Password" class="form-control input-md" required="" value=""</pre>
            onChange="validPassword(this.value)">

                 ript>
function validPassword(val) {
  var check = /(?=.*\d)(?=.*[a-z])(?=.*[a-z])/g;
  if (check.test(string(val))==true && val.length > 8) {
      document.getElementById("PasswordCheck").value = true;
      document.getElementById("PasswordCheck").innerHTML = "No errors";
}
}
                             document.getElementById("PasswordCheck").value = false;
document.getElementById("PasswordCheck").innerHTML = "Passwords must contain at least 8 characters and include uppercase, lowercase and
numbers";
            </script>
      </div
 <!-- Adding Re Enter Password and Verifying It-->
 <div class="form-group">
  <label class="col-md-4 control-label" for="ReEnter Password">Re-enter Password</label>
  <div class="col-md-4">
             <cinput id="ReEnterPassword" name="ReEnter Password" type="password" placeholder="Repeat Password" class="form-control input-md" required="" value=""</pre>
             onChange="validReEnterPassword(this.value)":
cp id="ReEnterPasswordCheck">
                        var Password = document.getElementById("Password");
if (Password.value == val) {
    document.getElementById("ReEnterPasswordCheck").value = true;
    document.getElementById("ReEnterPasswordCheck").innerHTML = "No errors";
                               document.getElementBvId("ReEnterPasswordCheck").value = false:
                               document.getElementById("ReEnterPasswordCheck").innerHTML = "Passwords must match";
             </script>
       </div>
 </div
```

As seen from the code snippets, the passwords were again checked using regex to include uppercase letters, lowercase letters and numbers. After that, the whole string was compared with the one typed in the re enter section and accept it if it is correct.

Following that, I have included the First Name and Last Name section which are identical. This is shown in the code snippets below:

```
<!-- Adding First Name and Verifying It-->
<div class=
    / class="form-group">
<label class="col-md-4 control-label" for="FirstName">First Name</label>
     <div class="col-md-4">
         <input id="FirstName" name="FirstName" type="text" placeholder="Enter First Name" class="form-control input-md" required="" value=""</pre>
         onChange="validFirstName(this.value)">

              document.getElementById("FirstNameCheck").value = false;
                        document.getElementById("FirstNameCheck").innerHTML = "First name should contain only letters, apostrophes, spaces and hyphens";
          </script>
     </div>
</div
<!-- Adding Last Name and Verifying It-->
     r class="form-group">
<label class="col-md-4 control-label" for="LastName">Last Name</label>
         <input id="LastName" name="LastName" type="text" placeholder="Enter Last Name" class="form-control input-md" required="" value="" onChange="validLastName"
</pre>
         (this.value)">
id="LastNameCheck">
         <script>
             function validLastName(val) {
    var check = /(^[A-Za-z \ ']+$)/g;
    if (check.test(string(val)) == true) {
        document.getElementById("LastNameCheck").value = true;
        document.getElementById("LastNameCheck").innerHTML = "No errors";
}
                  } else {
                      document.getElementById("LastNameCheck").value = false;
document.getElementById("LastNameCheck").innerHTML = "Last name should contain only letters, apostrophes, spaces and hyphens";
         </script>
     ¿/div
</div
```

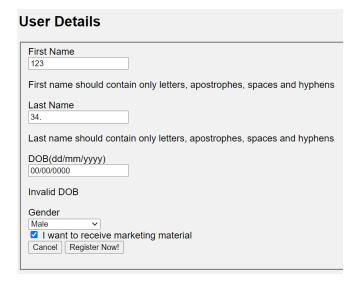
Again, names are checked using regex to only contain letters, apostrophes, spaces, and hyphens.

Lastly, there is the checking of the Date of Birth using the code snippet below:

```
<!-- Adding Date of Birth and Verifying It-->
cdiv class="form=group">
clabel class="form=group">
clabel class="form=group">
clabel class="col-md-4">
clabel class="col-md-4">
comed-4 control-label" for="DOB" >DOB(dd/mm/yyyy)</label>
cdiv class="col-md-4">
cinput id="DOB" name="DOB" type="text" placeholder="Enter DOB" class="form-control input-md" required="" value="" onchange="validDOB(this.value)">
cinput id="DOBCheck">
cinput id=
```

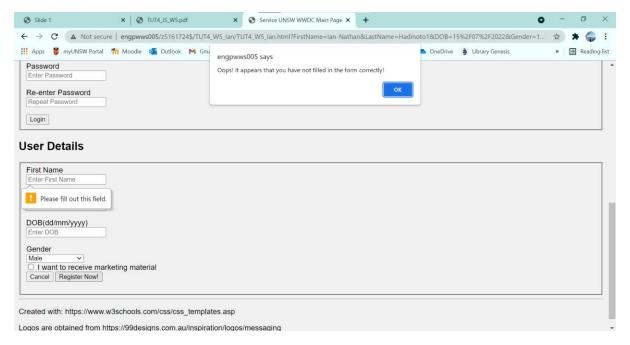
This was checked using regex to ensure that the Date of Birth was a valid date that was not too far in the past or in the future. To add on, these two screenshots below show the results when there is an input. The one on the left shows inputs with no errors and the one on the right shows incorrect inputs.



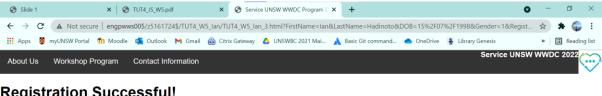


When the forms are accurate, the user can then hit the "Register Now!" button and the system checks if all their inputs are accurate. This is done using the code snippet below:

If there are some errors in the form filling in, the following alert will be displayed on screen along with a toolbox showing to fill in a missing detail:



If the details, the First Name, Last Name and Date of Birth, are all "true", the webpage changes to show the successful registration, as seen below:



Registration Successful!

Created with: https://www.w3schools.com/css/css_table.asp

Logos are obtained from https://99designs.com.au/inspiration/logos/messaging