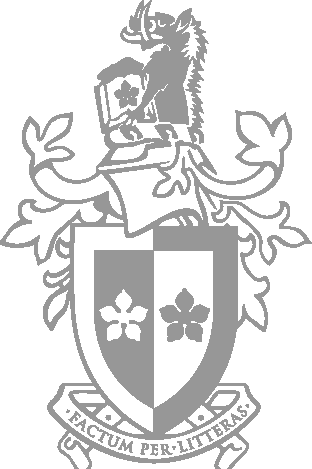
Faculty of Science, Engineering and Technology





**Interface Design and Development**

Distinction Task 3: My Registration Form

**Overview**

The registration is a process that is required when signing up for a web service.

.

**Purpose:** Demonstrate the use of Bootstrap and AngularJS form validation support, and the ability to implement a web registrations application.

**Task:** Create a registration form that is styled with Bootstrap and validated using AngularJS.

**Time:** This task should be completed before the start of week 9, but may be completed later (see the following note).

**Resources:**

* + - Lecture notes

**Note:** If you are not currently up to date you should skip this task and return to it once you are up to date with the Pass & Credit Tasks. Do not allow Distinction Tasks to delay you in keeping up with the unit's Pass and/or Credit Tasks.

***Submission Details***

You must submit the following files to Canvas:

* Web page source code (membership.html)
* AngularJS module (regmodule.js).
* Achecker report
* Screenshot of the table webpage

Make sure that your task has the following in your submission:

* Code must be HTML5 standard and observe indentation.
* The report shows no / as little-known problems as possible (membership.html)
* Your webpage uses the AngularJS and Bootstrap framework appropriately.



**Instructions**

Create a **registration web** form that uses **Bootstrap and AngularJS to validate and show error** messages. It must **prevent** the form from being submitted if errors still exist. There is no prescribe style for the form, however related input must be grouped accordingly using fieldset if necessary. The following are fields that must be in the form

1. First name (letters only and required)
2. Last name (letters only and required)
3. User name (min 3 characters and required)
4. Password (must contain at least 1 special character, minimum 8 characters and required, special characters are $, %, ^, &, and \*
5. Confirm password (must match above password)
6. Email (must be in email format and required)
7. Street address (maximum 40 characters, optional)
8. City (maximum 20 characters, optional)
9. Postcode (exactly 5 numeric digits and required, note some postcodes starts with 0)
10. Mobile number (10 digits, must start with 60)
11. Submit button (form data is submitted to the server only if the form data are valid)
12. Show Terms and Condition button (hides/unhides a brief text on terms and condition, random text is acceptable)

**Note:** You can test the form data submission by posting the data to previous Week 1 Lab formtest.php. Use the following form attribute and values for testing.

method="post" action="formtest.php"  
Or  
method="post" action="http://mercury.swin.edu.au/it000000/formtest.php"

All form data must be displayed with its corresponding value when posted.

Use the Accessibility Validator <http://achecker.ca> to test the form by ‘HTML File Upload’, choose the file, expand the ‘Options’ select ‘**WCAG 2.0 (Level AA)**’ and then ‘Check It’. Make sure the report shows no / as little known problems as possible (graded)

Your web app should now be complete. Make sure you test it on the browser to make sure that it works as you expect. Comment your code accordingly.