

## Hello World Assessment - WEB Frontend

*Write a program that prints “Hello, {UserName}” on the screen. {UserName} should be entered on a form by the user. If the userName starts with the letter “A”, the message should be “Hello, {UserName}, welcome back!”*

- To be completed using **AngularJs** only.
- Using object oriented principles, structure your code so that it is easy to **read**, **understand** what it does, **maintain**, and **reuse**, apply **DRY** principles. Your code should be readable without the use of comments.  
(e.g. `isColourBlue(colour)`)
- Set up a mock **RESTful** service showing:
  - posting inputted values back to the service.
  - retrieving data back from the service.
- For bonus:
  - Using **TDD**, create the spec to define the required functionality and tests your program.
  - Show off, show your knowledge of the language or framework and their features (e.g. mix-ins, prototyping, factories, directives, etc) and apply theory you have learned in the past (patterns, algorithms, programming structures, etc)
  - Use AngularJs
  - style using Twitter Bootstrap.
- The completed assessment should be published on the web, you can create a free account on Azure (preferable), AppHarbor, Heroku or Amazon web services.
- A link to your source code repository should also be made available - github, bitbucket are good.

**TDD** - Test driven development, part of the agile approach, used together with a CI (continuous integration) server. Here the spec/tests need to be written and runnable in the IDE used.

**ORM** - Object-relational mapping (ORM, O/RM, and O/R mapping) in computer science is a programming technique for converting data between incompatible type systems in object-oriented programming languages. This creates, in effect, a "virtual object database" that can be used from within the programming language. Our preference is a micro orm called ORMLite, supplied by ServiceStack.

**DRY** - Don't repeat yourself. self explanatory. No code should be repeated, algorithms structures and concepts should be encapsulated into class objects that can be easily reused, tested and maintained.