# Create a password indicator

## Node.js/NPM Runnable project

Please make sure that you have NPM installed with node.js to run the project.

* Dependencies: **Node.js** v10 with **NPM**
* Steps: Once project files extracted go to project path and run the following commands
  + $ **npm I** // to install dependencies via NPM
  + $ **npm run build-dev** // custom command to run (npm start and npm run test concurrently)
  + Or simply run commands separately to run the app (**npm start** or **ng serve**) and run the unit test using (npm run test or ng test)

## Mission

small Angular 5 project which contains a password indicator. The password indicator may display the strength of a password. Built as reusable component with its own styles.

## Applied Requirements

* The password strength should be displayed as a bar with 12 units.
* The highlighted colours of the first 4 units are red,
* the next 4 units are orange and the last 4 units are green.
* Initial colour of each unit is grey.
* Mock an API call to simulate a remote password strength calculator which returns digits between 1 and 12.
* Wire up the mock service like a remote API call.
* Feel free to define the rules for the mock service to calculate the password strength.
* Add an input field which calls the API on every keypress.
* Minimum password length should be 6 characters. The aforementioned bar have to highlight the units up to the returned value.
* Additional installed 3rd party libraries are totally free licensed.
  + **ng-bootstrap**: *to support ready UI components (tooltip, validation alerts)*
  + **bootstrap4**: *to support UI dev and responsive design*.
  + **@jasmine/types**: *to support jasmine types (describe, expect, …)*
  + **Concurrently** [*https://www.npmjs.com/package/concurrently*](https://www.npmjs.com/package/concurrently)

*To apply concurrent scripts in one command-line to build custom command line*

***npm run build-dev***

* ~~use of jQuery or similar DOM changing JavaScript frameworks is not included~~
* related to the Angular 5 style guide conventions for naming objects and project structure
* deployment will be done by **webpack**: *tested with* ***webpack 3.11.0***
* **Applied Unit test:** unit-tests for existing components and service with some of unit test cases (**not covered 100%**)
* **Responsiveness:** Tryed to apply responsiveness as I could related to the time.

## Tech. environment

Basics: HTML 5, CSS3  
Programming language: Typescript, version: 2.9.2  
Frameworks: Angular 5.2.11, CLI: 1.7.4, Bootstrap 4

## Miscellaneous

followed Angular 5 style guide conventions for naming objects, organize etc. this project.

Notice that the later deployment will be done by webpack so some component selectors e.g. moduleId are not needed.

## Time frame

**Real programming time:** *6 – 7 hours*

**Total time:** 10 hours

* **Structure and architecture:** *1 hours*
* **Development environment installation:** *1 hours*
* **UI-Design & implementation:** *2 hours*
* **Write and Implement strength algorithm:** *2 hours*
* **Unit**-**test config and implementation:** *2 hours*
* **Other issues:** *2 hours*

***Thanks a lot for reviewing my documents, I hope to hear from you soon.***

***Khallad Sharaf El-Din***