# Password Service

## Description

When defining passwords for an account, it's often useful to tell the users the level of security of the passwords they choose.

The goal of this challenge is to implement a system that helps our users select a good and strong password.

### Objectives:

- Implement a backend service that will receive the password and return either ***WEAK***, ***OK*** or ***STRONG*** based on specific password rules
- Implement a frontend application that calls the back-end service to measure the password and show the user the strength of their password as they are typing
- A ***STRONG*** password passes all 6 rules
- ***WEAK*** and ***OK*** passwords are at your discretion

### Required Rules:

1. Should have minimum length of 6 characters
2. Should have lower case and upper case words
3. Should have at least 1 number
4. Should have at least 1 symbol
5. Should not contain sequence characters (eg. "aaaa", "w@@w", "oo11")
6. Should not be part of a blacklisted password list (see file appendix *password_blacklist.txt*)

*NOTE: other rules might be required in the future*

## Implementation Notes

- Acceptable implementation languages are Ruby and Javascript. You may use any language framework readily available to you (using Sinatra and React is encouraged)
- Completed projects must include a README with enough instructions for evaluators to build and run the code
- Use appropriate production-capable frameworks
- Use appropriate dependency-management and build tools
- The project's structure and organization should follow best practices
- Prefer immutable design if possible
- Test your code and your API. No need to test every permutation, but demonstrate you know the types of things to test for.
- Even though this is a simplified requirement as appropriate to be an exercise, your code should be production capable
- Show your working, if you've used any interesting libraries or approaches during development let us know and explain why in the README