

# Agreements & Def of Done & Style Guide

## Team Agreements:

- Meet MWF for scrum meeting @10:45-11am
- Meet MWF for group working sessions @11am-12pm
- If you need to be on zoom for any particular reason, please let everyone know ahead of time by at least 1 hour
- Main communication will be through discord group chat
- If it is an emergency, then message in the text group chat
- If you are stuck on a task, immediately ask for help. Someone may know a solution right away

## Definition of Done:

- A feature must have all functions outlined by its user story specification
- The feature must work for all test cases including edge cases (manual testing)
- The code for the feature must conform to the style specification
- The feature must be merged into the main branch without any conflicts
- The feature must be integrated into the app without consequences to other features and such that it is of the same visual style as the other features

## Style Guide

### **Repository/Project Structure**

#### Frontend

- COMPONENTS/
  - JS files that make up part of a page, ie buttons, appBar, goal setting, etc.
- screens/
  - JS files that is displayed to a user
- ASSETS/
  - Example images outputted from backend to be display to user (mainly used for graphing functionality)

#### Backend

- INTERFACE/
  - Files to handle database fetching directly

## Coding Guide (general JS formatting)

### General Guidelines

- Avoid magic literals
- Avoid code duplication
- Avoid negative conditionals
- Use explanatory variables/functions (simple and purposeful)
- Use constants instead of magic numbers
- Functions should do one thing, name should be self explanatory
- Define variables and functions close to where they are used
- Define private functions just below its usage

### Programming Guidelines

- Spacing
  - Indentation: 4 spaces
  - 1 space in between all operators i.e (x + 1), not (x+1)
  - Limit horizontal spacing
  - 1 vertical space in between functions
  - 1 vertical space in between imports and functions
  - 1 space after a (:)
  - 1 space after a (,) or (;), not before
  - No space at start or end of functions
- Comments/JSDoc
  - `/* */` for multi line,
  - `//` for single line use
- Language
  - Use const (default) or let, do not use var
  - Use trailing commas

Example:

```
const values = [  
  'first value',  
  'second value',  
];
```

- Don't use `Array(x1)`, use `[x1]` instead
- Don't use `Object(x1)`, use `{x1}` instead
- Subclasses must call `super()` before accessing or setting fields

- All imports should be defined at the top of a file
- Nested functions should be defined by a local const
- Use single quote for string literals rather than double quote ('template')
- Template literals using (`\${template}`)
- No line continuations, line length should be 80 (this can be roughly 80 BUT DO NOT EXCEED 95)

Example:

```
const longString = 'This is a very long string that far exceeds the 80 ' +  
    'column limit. It does not contain long stretches of spaces since ' +  
    'the concatenated strings are cleaner.';
```

- No empty catch blocks
- File Naming
  - File encoding in UTF-8
  - File names should be UpperCamelCase
  - Files should end in .JS, unless otherwise specified
- Naming
  - Class names should be UpperCamelCase
  - Methods names should be lowerCamelCase
  - Variable names should be in camelCase
  - Constants should be ALL\_CAPITALIZED