Inspection Issue Log

Project: <u>Automated Turbulence Detection</u>

Inspection ID: <u>INSPECTION1</u>

Meeting Date: <u>April 4, 2018 (Wed)___</u>
Recorder: <u>___Quyen Tran_____</u>

Origin:	**R**equirements, **D**esign, **C**onstruction, **T**esting
Type:	**M**issing, **W**rong, **E**xtra, **U**sability, **P**erformance,
	Style, **C**larity, **Q**uestion
Severity:	**M**ajor, **m**inor

14 Defects Found: 3 Major, 11 minor   **Defects Corrected: <u>3</u> Major, <u>11</u> minor**

	Origin	Type	Severity	Location	Description
1.	C	W	M	weather_fun.py	Within the algorithm to predict weather, a hard coded year of 1800 is used as the starting point. It appears that data accuracy would be improved by querying data of the current year through using a different constructor.
2.	C	W	m	weather_fun.py	At line 24-25, index_1_reg is checked to be not none before replacing it by the argument passed in. However, this is incorrect and may lead to the index_1_reg not being correctly replaced. Additionally, the if clause will not be satisfied because when the init method is called, rather, it would always be None.
3.	C	Q	m	Statistics.py	On line 40, 51, 62 and 74 for storing data, it appears that new line identifier is passed in as an empty string. This may work on platforms such as Windows('win32'), but this may break under Linux or macOS('darwin').
4.	D	S	m	Simulator.py	Imports should be done in a more organized manner. For example, imports from the standard library can be declared first, while external imports are declared next and self-written classes can be imported last.
5	D	C	m	Simulator.py	"dif_a" and "dif_b" (line 221-222) are somewhat deceptive and confusing names. "diff_end_minus_start" and "diff_start_minus_end" would clearly indicate their purposes.
6	D	C	m	models.py	Magical numbers (line 9, 13, 15-17, 24-26) are used for truncating decimal places, it can be clearer if comments are given to justify these numbers.
7.	D	E	m	views.py	Within line 1-13, timedelta has been imported twice from datetime. Duplicate imported symbols can lead to performance issues.

8.	C	C	m	admin.py	The file does not seem to serve any purpose other than importing from two libraries, and nothing else seems to refer to it. Is the file a part of future implementations and does the file actually need to exist?
9.	T	M	M	tests.py	The test file is empty, signifying there are no unit or UI test cases for the software. Are there plans to add tests in the future to ensure code quality?
10.	D	C	m	weather_data.js	This entire file contains an array of location points and weights, however it is unclear of how are the weights are determined, and the project can be more maintainable if the algorithm used to calculate or generate weights are included in the project.
11.	D	C	m	index_script.js	There are two hard coded coordinates on line 6 and 177 to be used for the center position of the map. It would be useful to at least have comments as of why those coordinates are chosen.
12.	D	C	m	main.js	At line 180, magical numbers 8 and 0.9 are used as scaling factors when calculating bounds, while their purposes are unclear. It may be helpful to declare them as constants, or to calculate them algorithmically.
13.	C	M	M	index_css.css	At line 19, 47, 72, 86, 100 and 109, you have repeatedly declared the font-family as Roboto. However, since no web font has been embedded and Roboto is not pre-installed on major desktop operating systems, it is necessary to declare fallback fonts. A potential suggestion is to use the system-ui declaration, so that it uses system font on all platforms.
14.	D	E	m	index_css.css	On a related note to issue 13, there are duplicate, identical font entries at line 19, 47, 72, 86, 100 and 109. It would be far more maintainable to consolidate them into one entry under the body or html tag.