Test Planner and Tracker

Team 30 **Rishabh Daga, Anubhav Sharma, Monil Gokani, Shiva Prasad**

Test No. ID	Related Use case	Pre-conditions	Test Description (steps)	Expected Outcome	Sprint 2 Outcome (color code cell background)	Comments (if test case failed)	Sprint 3 Outcome (color code cell background)	Comments (if test case failed)	Sprint 4 Outcome (color code cell background)	Comments (if test case failed)	R1 Outcome (color code cell background)	Comments (if test case failed)
1	Check if the UI is working properly.	The website should open with all the components loaded and the buttons and dropdowns working correctly.	Check if the website is working properly and the references and links should be well connected. The working of links could be checked by looking at the addresses of the links.	All webpages should open on clicking on their links. The submit buttons and dropdowns should be working correctly		As sprint 2 did not involve any coding related to UI.and in general anything .		As sprint 3 did not involve any coding related to UI.and in general anything .				
2	Check if the data is correct or not.	Raw data must be available on the Thingspeak server.	Check by looking at random points in data manually and then running a script to take care of the irregular and impossible values in the data.	Correct data with no abnormalities.		As sprint 2 did not involve any coding related to UI.and in general anything .						
3	Check sample graphs which are plotted.	Raw data must be available on the Thingspeak server.	Check by looking at the data (some data points are logicaly absurd). Running a check script in python which will detect any abrupt changes.	Line graphs and Histograms with no abrupt/illogical jumps.		As sprint 2 did not involve any coding related to UI.and in general anything .						
4	Check scripts coded for Line graphs.	Raw data should be available on the Thingspeak server. The script should be available and running at the backend for further processing of any request from front end.	By comparing the line graph generated and the data used from the server.	Line graphs in accordance with the data. Patterns could be inferenced from it proving the correctness of itself.		As sprint 2 did not involve any coding related work.						
5	Check scripts coded for Histograms.	Raw data should be available on the Thingspeak server. The script should be available and running at the backend for further processing of any request from front end.	By comparing the histograms generated and the data used from the server.	Histograms in accordance with the data. Patterns could be inferenced from it proving the correctness of itself.		As sprint 2 did not involve any coding related work.						
6	Check the storage of static data stored in SQL databases at the backend.	The system should be online. The server should be up and running.	By running a SQL query in the database where the data is stored.	Correct output should be returned.		As sprint 2 did not involve any coding related work.		Did not create a framework and database by then.				
7	Check the functionality of backend.	The code at backend should be working. The server should be up and send data regularly.	Not yet decided how to implement.	Output should match the correct ouput produced manually.		As sprint 2 did not involve any coding related work.		As backend not yet developed.		As backend is not a requirement for R1.		
8	Check if the interpolation done is correct or not.	The website should open and display graphs for the fields entered.	By looking at the interpolated graph and the neighbouring correct values of the incorrect data from the dataset.	The graph should not vary much and should be consistent with the remaining correct data.		As sprint 2 did not involve any coding related work.		As we are not using live data yet so the level of interpolation is not high.		As we are not using live data yet so the level of interpolation is not high.		
9	Check if the conclusions drawn are correct or not.	For this we should basically have data over which we can draw conclusions	Looking at the trends in our graphs and checking for possible explanations for their behaviour.	Output should correspond to an occurence acting as a reason for the corresponding behaviour of the graphs.		As sprint 2 did not involve any coding related work.		Need more data to make correct conclusions.				