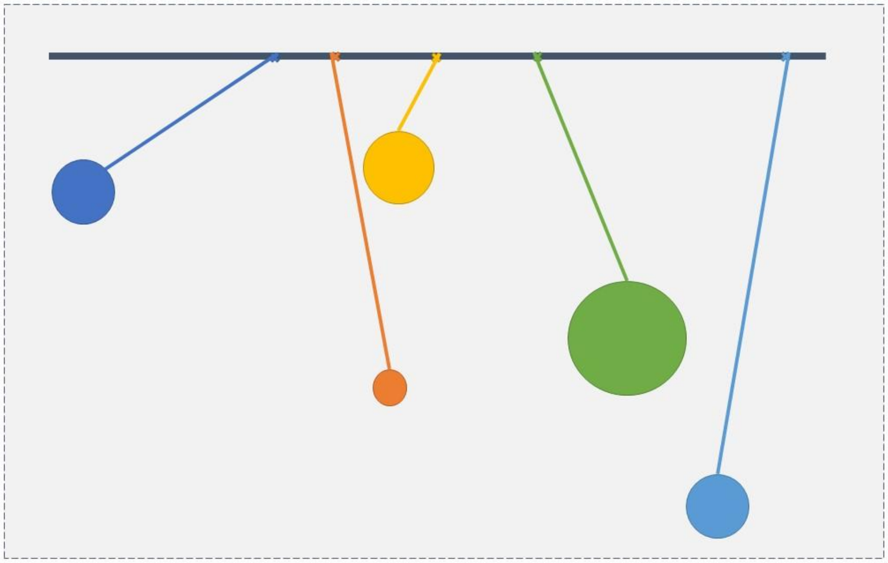
Take Home Exercise

In node.js, implement a "simple pendulum" in 1 dimension. Let there be some random "wind" (in 1 dimension) agitating the pendulum. A REST interface will allow to set up an initial angular offset, a mass, a string length, a maximum wind factor (or other settings of your choice). It will also allow you to read the last coordinates, such as to be able to represent the pendulum in a UI, in an external process. The rate of simulation may be higher than the rate of visualization. Let's run 5 instances of the simple pendulum node.js process, each on its own tcp port. During configuration, also tell each node who are their immediate neighbours.

Let's have one UI process, displaying in the web browser the five pendulums. It would allow the user to configure them (starting angle, wind factor, mass, length, or anything else you have chosen) in an intuitive way, conveying an easy-to-use user experience. The UI should expose a canvas where the user could drag and drop the pendulum in their desired start position. The UI should be done using pure javascript and some CSS. Finally, all the resulting parameters of that "configuration" would be transferred in JSON format to the REST API.



The UI would also expose some simulation controls (start, pause, stop), and would poll periodically the 5 pendulum instances (using a HTTP client) to display them. A refresh rate of a few frames per second is fine. During the simulation, each pendulum node should watch the position of its immediate neighbours periodically (via REST). Whenever they come "too close" (threshold is at your discretion), send a STOP message using a "guaranteed delivery" communication channel of your choice, to all 5 instances. Upon reception of STOP, each instance stops moving immediately, and waits for 5 seconds before sending RESTART on the same channel. Once all 5 instances have received all 5 distinct RESTART messages, all will restart moving, jumping back to their original position.

Document briefly the resulting REST interface.

We hope you enjoy the challenge!