The Node Manager is a full-stack development problem, involving a simulated scenario. Once the primary criteria is met, we hope you’ll have some fun in implementing some extra touches to the application. If you have any questions, or require clarification, please don’t hesitate to e-mail us.

## Submission Notes

* Send in your submission as a ZIP file, with the name format: <FirstName>.<LastName>.zip
* Ensure the Primary Criteria runs without error.
* Employ best practices in your design and coding.
* While the problem is full-stack, we recommend focusing on areas which you are most proficient in
* Sample code and unit tests have been provided for context. You may choose to use or discard as you wish.
* Our preference is for a .NET implementation, but feel free to use the language/platform of your choice.
* If your submission is not in .NET, please provide any applicable build/execution instructions.
* Include a description of your submission, by filling in the sections at the end of this document.
* Third Party APIs, libraries, frameworks and services may be leveraged (e.g., Google, Bing etc.,), but they must be legal, and accessible by us.

## Background

Nodes are fictional services deployed in datacenters across the country, which are connected to each other to share important data. These nodes also report important telemetry, providing metrics and information about their status.

## User Story

Our Network Operations team needs an application which will allow them to monitor and manage various nodes within our network.

## Primary Criteria

* Ensure the user can add/remove nodes to the application, when a node is deployed, or removed from a data center.
* Ensure the user can bring a node online, or take it offline.
* Ensure the user can determine whether or not a node is online, or offline
* Ensure the user can see telemetry being reported from a node
* Ensure the user can set maximum threshold values for node telemetry metrics.
* Ensure an alarm is presented to the user when any of the metrics exceed maximum limits.

## Bonus suggestions

*Here are some suggestions for creating an outstanding submission.*

* Have the metrics update automatically
* Demonstrate defensive coding and error handling.
* Demonstrate good use of design patterns
* Consider the intuitiveness of the UI
* Visualize the nodes on a map
* Increase unit test code coverage
* Expose the application logic as a SOAP or REST service
* **Anything else: Surprise us :)**

## Briefly describe the design of the submission. Are there any particular areas we should focus on?

## Were any bonus suggestions implemented?

## Are there any items which are still in-progress?

## Are there any specific instructions required to execute the application?

## Is there anything else we should know about the submission?