Setup Guide for Monitoring and Alerts using Application Insights for Node.js/React

This document provides a detailed guide on how to set up monitoring, metric tracking, error detection, and alerting using Application Insights in a Node.js/React application.

# 1. Setting Up Application Insights for Node.js

Step 1: Install Application Insights SDK:

npm install applicationinsights

Step 2: Configure Application Insights in your Node.js application:

const appInsights = require('applicationinsights');  
  
appInsights.setup('<INSTRUMENTATION\_KEY>')  
 .setAutoDependencyCorrelation(true)  
 .setAutoCollectRequests(true)  
 .setAutoCollectPerformance(true)  
 .setAutoCollectExceptions(true)  
 .setAutoCollectDependencies(true)  
 .setAutoCollectConsole(true)  
 .setUseDiskRetryCaching(true)  
 .start();

Replace <INSTRUMENTATION\_KEY> with your instrumentation key.

## Example Node.js Application:

const express = require('express');  
const app = express();  
  
app.get('/', (req, res) => {  
 res.send('Hello World!');  
});  
  
app.get('/error', (req, res) => {  
 throw new Error('This is a test error');  
});  
  
const server = app.listen(3000, () => {  
 console.log('Server is running on port 3000');  
});

# 2. Setting Up Application Insights for React

Step 1: Install the Application Insights npm package:

npm install @microsoft/applicationinsights-react-js @microsoft/applicationinsights-web

Step 2: Configure Application Insights in your React application:

import { ApplicationInsights } from '@microsoft/applicationinsights-web';  
import { ReactPlugin } from '@microsoft/applicationinsights-react-js';  
  
const reactPlugin = new ReactPlugin();  
const appInsights = new ApplicationInsights({  
 config: {  
 instrumentationKey: '<INSTRUMENTATION\_KEY>',  
 extensions: [reactPlugin],  
 extensionConfig: {  
 [reactPlugin.identifier]: { history: browserHistory }  
 }  
 }  
});  
appInsights.loadAppInsights();

Replace <INSTRUMENTATION\_KEY> with your instrumentation key.

## Example React Application:

import React from 'react';  
import ReactDOM from 'react-dom';  
import { withAITracking } from '@microsoft.applicationinsights-react-js';  
import { reactPlugin, appInsights } from './AppInsightsConfig';  
  
class App extends React.Component {  
 render() {  
 return (  
 <div>  
 <h1>Hello, World!</h1>  
 <button onClick={() => { throw new Error('Test error'); }}>Throw Error</button>  
 </div>  
 );  
 }  
}  
  
const AppWithTracking = withAITracking(reactPlugin, App);  
  
ReactDOM.render(<AppWithTracking />, document.getElementById('root'));

## AppInsightsConfig.js:

import { ApplicationInsights } from '@microsoft/applicationinsights-web';  
import { ReactPlugin } from '@microsoft/applicationinsights-react-js';  
  
const reactPlugin = new ReactPlugin();  
const appInsights = new ApplicationInsights({  
 config: {  
 instrumentationKey: '<INSTRUMENTATION\_KEY>',  
 extensions: [reactPlugin],  
 extensionConfig: {  
 [reactPlugin.identifier]: { history: window.history }  
 }  
 }  
});  
appInsights.loadAppInsights();  
  
export { reactPlugin, appInsights };

# 3. Setting Up Alerts in Application Insights

Step 1: Sign In to Azure Portal.

1. Navigate to the [Azure Portal](https://portal.azure.com/).

2. Sign in with your Azure account.

Step 2: Create Alerts in Application Insights:

1. Navigate to your Application Insights resource.

2. Select "Alerts" from the left menu.

3. Click "New alert rule".

4. Choose a condition, such as the number of errors exceeding a certain threshold.

5. Select an action group to send alerts via email, SMS, webhook, etc.

6. Name the alert and click "Create alert rule".

# 4. Testing and Verification

1. Run your Node.js and React applications.

2. Check the Azure Portal to see if logs and metrics are being sent to Application Insights.

3. Trigger test errors to see if alerts are activated.

# Conclusion

With these steps, you have successfully set up a monitoring and alerting system for your Node.js/React application using Application Insights. If you have any questions or need further assistance, do not hesitate to reach out.