



Event Generator (AWS Simulator)

Programmer's Guide

Table of Contents

I.	Summary	2
II.	Source Files	3
III.	Circumstances of the Software	4
IV.	How to Build and Run the App Locally.....	4
V.	How to Deploy the App to Heroku.....	5

Summary

The Cisco Dynamic RevProxy team's Event Generator simulates the data AWS (Amazon-Web-Services) is configured to send (See. [AWS_CloudWatch_Instance_Monitoring_Guide.pdf](#)). These events consist of 'Run' and 'Terminate' calls on instances running on AWS. To simulate such events, the Event Generator sends HTTP requests with sample event data.

The instances and event data used in the software are derived statically for simulation purposes (typically this data would be dynamically created on AWS). The Event Generator sends HTTP requests to a server that simulates Cisco's Dynamic Reverse Proxy. Run statuses for each instance are displayed and synchronization techniques are used between the server and the event generator to keep them up to date.

The Event Generator is built using React and uses Facebook's 'Create-React-App'. The foundation for React-App can be found here:

<https://github.com/facebook/create-react-app>

Source code for the Event Generator can be found here:

<https://github.com/Will-Rooney/cseventgenerator>

The deployed Event Generator lives here:

<https://cseventgenerator.herokuapp.com/>

For more detailed information see:

<https://github.com/Will-Rooney/cseventgenerator/blob/master/README.md>

Source Files

1. Folder Structure

```
cseventgenerator/  
  README.md  
  node_modules/  
  package.json  
  public/  
    index.html  
  src/  
    App.css  
    App.js  
    App.test.js  
    Instance.css  
    Instance.js  
    Instance.test.js  
    InstanceList.css  
    InstanceList.js  
    InstanceList.test.js  
    IntervalSlider.js  
    RangeSlider.css  
    index.css  
    index.js  
    instance_list.json  
    runInstanceEvents.json  
    terminateInstanceEvents.json
```

2. Primary File Descriptions

- a. index.js
 - Loads the generated html from App.js into index.html.
- b. App.js
 - Renders the instance table from InstanceList.js.
- c. InstanceList.js
 - Utilizes IntervalSlider.js to adjust interval durations.
 - Renders each row of the instance table using the list of instances.

The data for each instance is generated from Instance.js.
Handles Child instance run statuses by querying the server.

d. Instance.js

Generates all table data for the current row (instance).
Handles individual HTTP (Automatic or Manual) requests that simulate
AWS instance events.

e. instance_list.json

Contains a static list of all instance names

f. runInstanceEvents.json

Contains a static list of all sample run instance events for each instance in
instance_list.json

g. terminateInstanceEvents.json

Contains a static list of all sample terminate instance events for each
instance in instance_list.json

Circumstances of the Software

The react application runs and builds successfully. The program was developed using JavaScript (JSX), compiled and ran using Yarn (in place of NPM). The program was tested on Windows 10 and Linux Mint. A deployed version of the program is on Heroku here:
<https://cseventgenerator.herokuapp.com/>

The program requires NodeJs, NPM and Yarn to build and run the application.

How to Build and Run the App Locally

To install all the required dependencies first open a terminal or command prompt and run the following command from the project root directory (cseventgenerator/)

```
yarn install
```

To create an optimized build of the application run the following command:

```
yarn build
```

A new directory 'build/' will be created. The build folder is ready to be deployed. You can deploy it with a static server:

```
npm install -g serve  
serve -s build
```

To run a local development version of the react application run the following commands in a terminal or command prompt:

```
yarn start
```

The react application will automatically open in your default web browser at <https://localhost:3000/>.

For further information see <https://github.com/Will-Rooney/cseventgenerator#available-scripts> and use 'yarn' in place of 'npm'.

How to Deploy the App to Heroku

To deploy to Heroku, a [Heroku](#) and [Github](#) account is required. Follow the steps listed here: <https://github.com/Will-Rooney/cseventgenerator#heroku>