### **Express Generator**

#### Objectives and Outcomes

In this exercise, you will use the Express generator to scaffold out an Express application. Thereafter you will modify the application to support REST API making use of the Node modules that you developed as part of the first assignment. At the end of this exercise, you will be able to:

* Generating an Express application using the express-generator
* Modify the Express application to support the REST API by adding routes

#### Installing express-generator

* Install express-generator by typing the following at the prompt:

     npm install express-generator

Use sudo if you are using an OSX or Linux machine.

**Note**: You may need to restart your terminal/cmd window in order for the express command to be available at the prompt.

#### Scaffolding an Express Application

* To scaffold out an Express application, type the following at the prompt:

     express conFusionServer

* Next, move to the conFusionServer folder. Type the following at the command prompt to install all the Node modules

     npm install

* You can start the Express server by typing the following at the prompt:

     npm start

* Add a file named .gitignore to the project folder and type the following into the file:

node\_modules

* Initialize a Git repository and do a Git commit with the message "Express Generator".

#### Implementing a REST API

* Now, copy the dishRouter.js, promoRouter.js and leaderRouter.js from your first assignment (node-express/routes folder) to the routes folder within the Express application that you just scaffolded out.
* Furthermore, copy the index.html and aboutus.html file from the node-express/public folder to the public folder in your new project.
* Then, open the app.js file and then update the code in there as follows:
* . . .
* var dishRouter = require('./routes/dishRouter');
* var promoRouter = require('./routes/promoRouter');
* var leaderRouter = require('./routes/leaderRouter');
* . . .
* app.use('/dishes',dishRouter);
* app.use('/promotions',promoRouter);
* app.use('/leaders',leaderRouter);
* . . .
* Save the changes and run the server. You can then test the server by sending requests and observing the behavior.
* Do a Git commit with the message "Express Generator REST API".

#### Conclusions

In this exercise, you learnt to scaffold out an Express application using the express-generator. Thereafter you were able to build a REST API server.