

# JSDT demo 01

Patrik Suzzi

Version 1.0, 23.10.2016

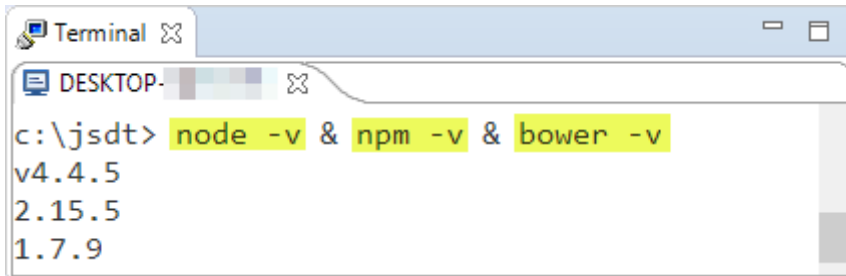
# JSDT demo 01

**Angular.JS** single page application with *routes* and *controller*. Styling is done via **Bootstrap** and **FontAwesome**. The demo also has a **Node.js** *script* to launch a local **Express** *server* to serve the application.

## Prerequisites

You installed latest Eclipse JSDT or JEE. As JSDT integrates Node.js (and its tools) on your system, you should also have installed Node.js, npm and Bower.

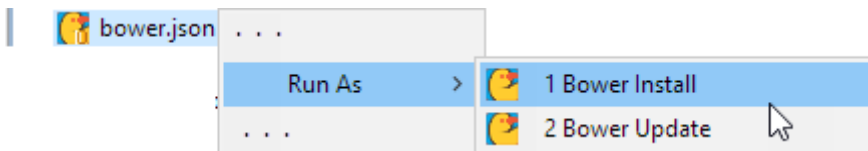
To verify the prerequisites, presse kbd:[Ctrl+Alt+T] to open the command prompt in Eclipse, and type **node -v**, **npm -v** and **bower -v**, to verify the tools are installed.



```
Terminal
DESKTOP-
c:\jsdt> node -v & npm -v & bower -v
v4.4.5
2.15.5
1.7.9
```

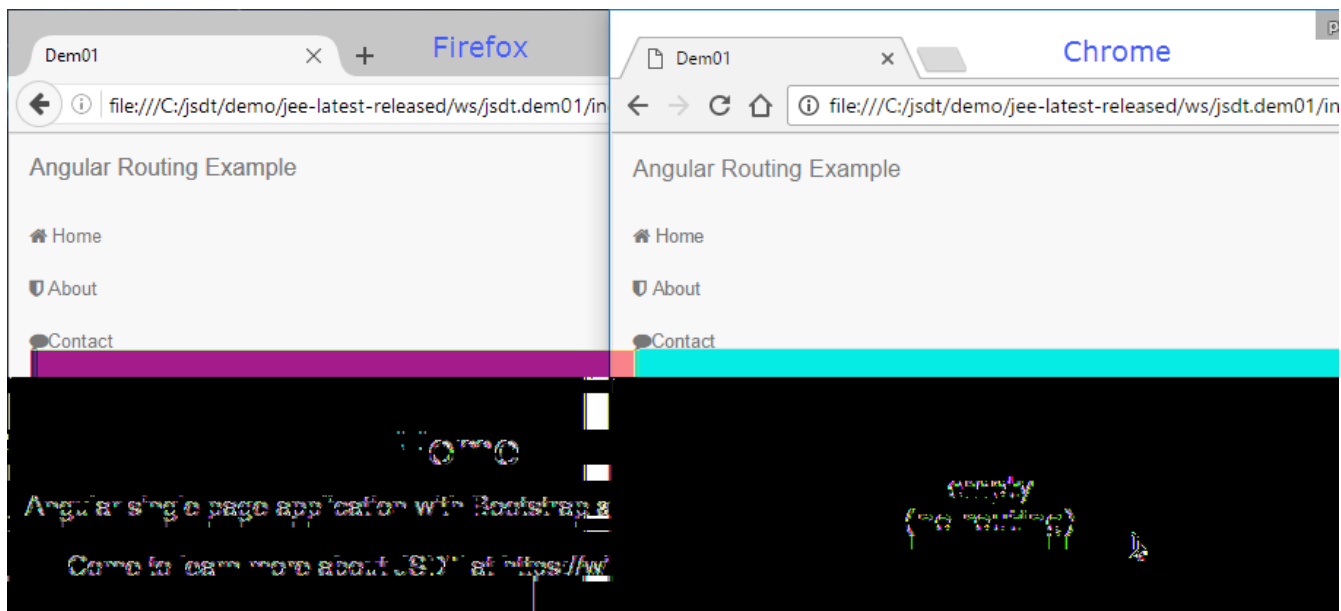
## Run the application

install bower dependencies: r-click on **bower.json** then select **Run As.. > Bower Install**. This will install the needed dependencies under the **bower\_components** folder



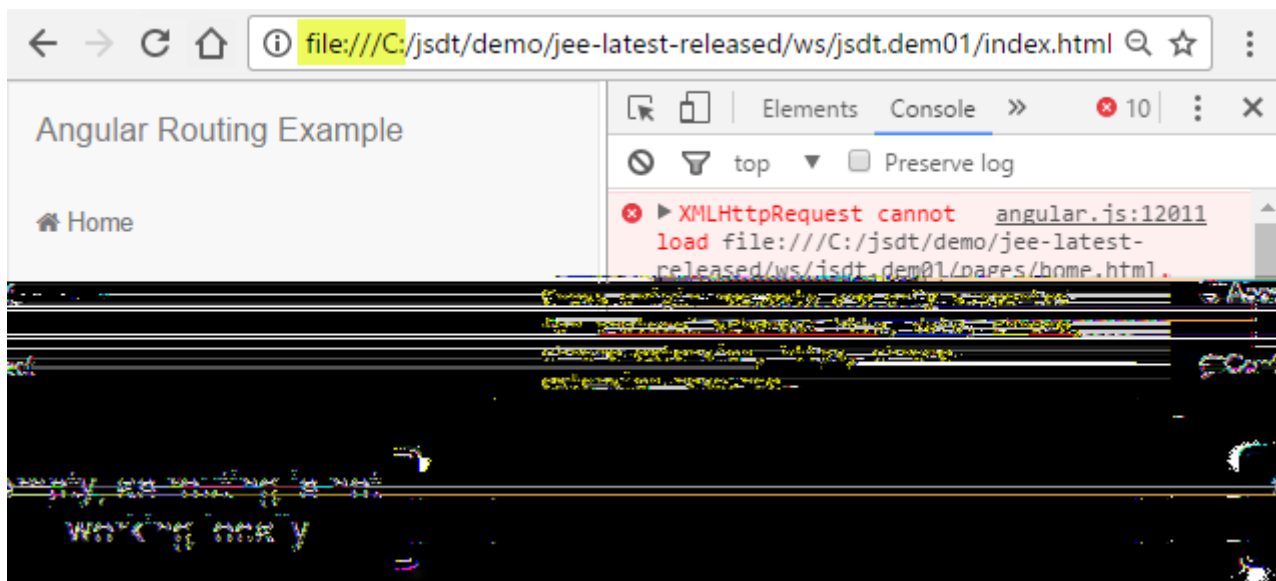
Now you can open *index.html* in your browser.

Actually, you'll see the application will work on Firefox but not on Chrome



## Cross-origins requests

Angular routing is not working on local files. Hence, you'll see an empty page when launching the page in Chrome. You can detect the error only by inspecting the page.



To fix this, we'll use nodejs to launch a lightweight local server to serve static content.

```
// define port
var port = 3222;

// serve static content
app.use(express.static(__dirname + '/'));

// launch server
app.listen(process.env.PORT || port, function() {
  console.log('Example app listening on port ' + port + '!');
});

// open with default browser
open('http://localhost:' + port);
```

To run the script, we need to install the needed libraries, and to run the script with nodejs

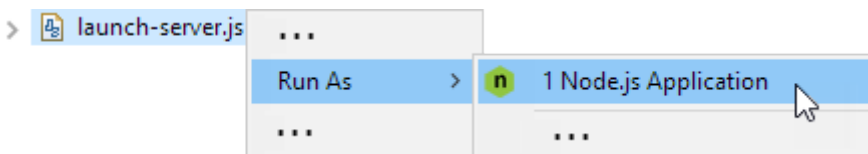
#### *install dependencies*

R-click `package.json` and select **Run As.. > NPM Install**. This will install the needed dependencies under the `node_modules` folder

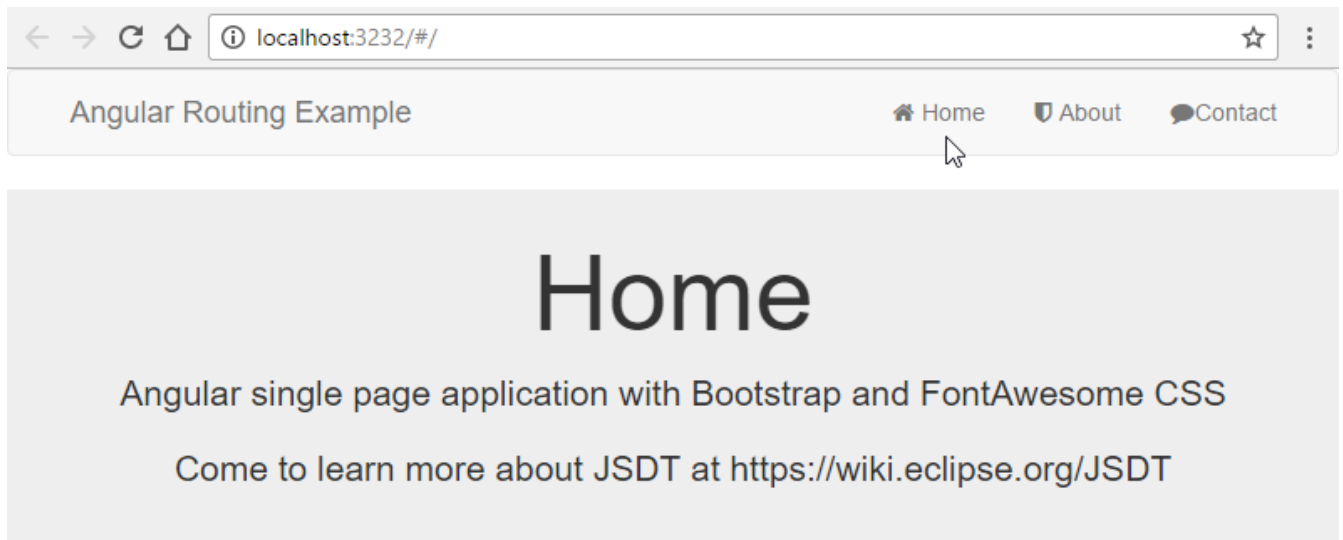


#### *run node script*

R-click 'launch-static-server.js' and select **Run As .. > Node.js Application**. This will launch node passing your script as parameter



The script launches the server, and loads the content of the current directory, which displays our `index.html`, where you can see the content is loaded using angular routing.



To stop the server, you can just terminate the process in Eclipse.

---

## Copyright & License

CopyRight © Patrik Suzzi, 2016. You can use the software examples free of charge, under the terms of [EPL](#) license.