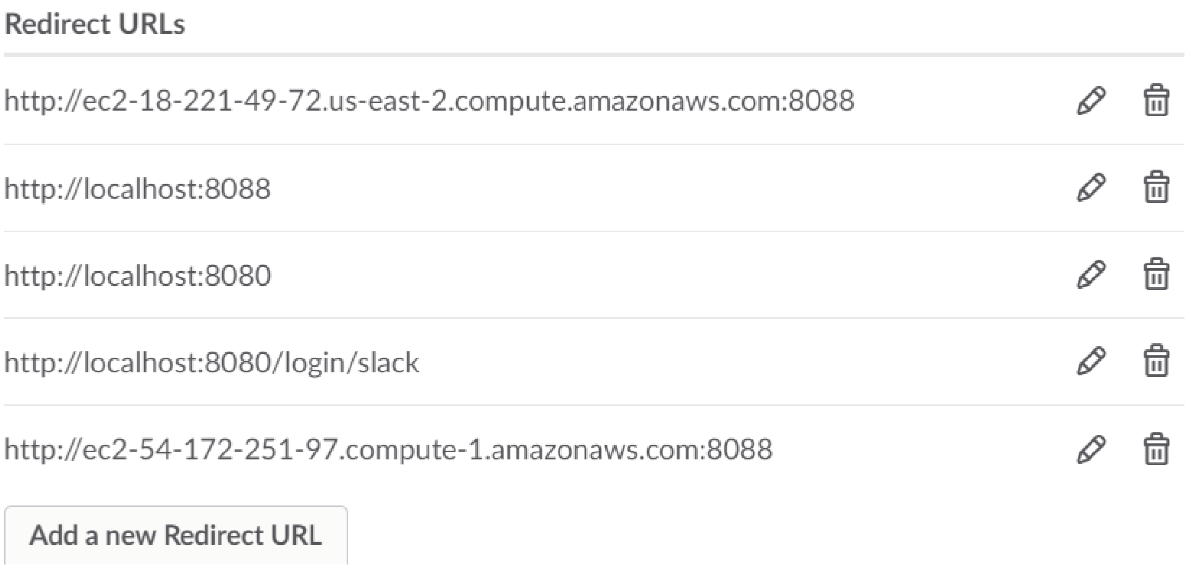
Running Rideshare locally

Step 1: Install nvm provided in Folder.

Step 2: install node.js and npm using Command Prompt

1. # Install Node.js and npm
    a. nvm install v6.11.1
2. # Get the latest version of npm
    a. npm i -g npm

Step 3: Check your application.properties file…Here you'll need to fill out Database and Slack information for an application to function fully. Also make sure localhost:port is enable in Slack API. (where you give your EC2 instance address.)



Redirect URLs

http://ec2-18-221-49-72.us-east-2.compute.amazonaws.com:8088

http://localhost:8088

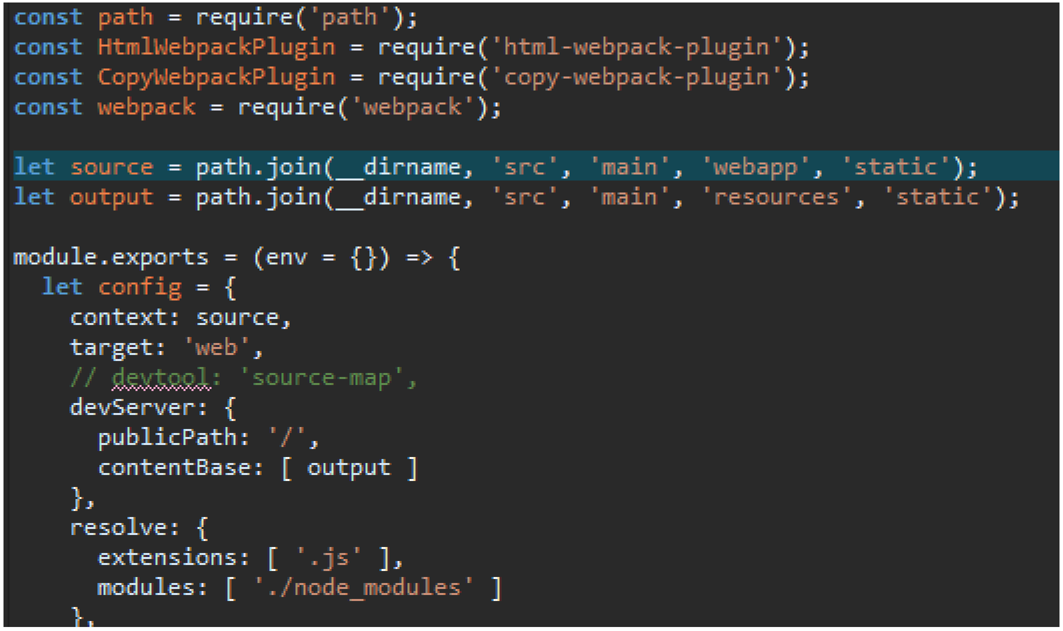
http://localhost:8080

http://localhost:8080/login/slack

http://ec2-54-172-251-97.compute-1.amazonaws.com:8088

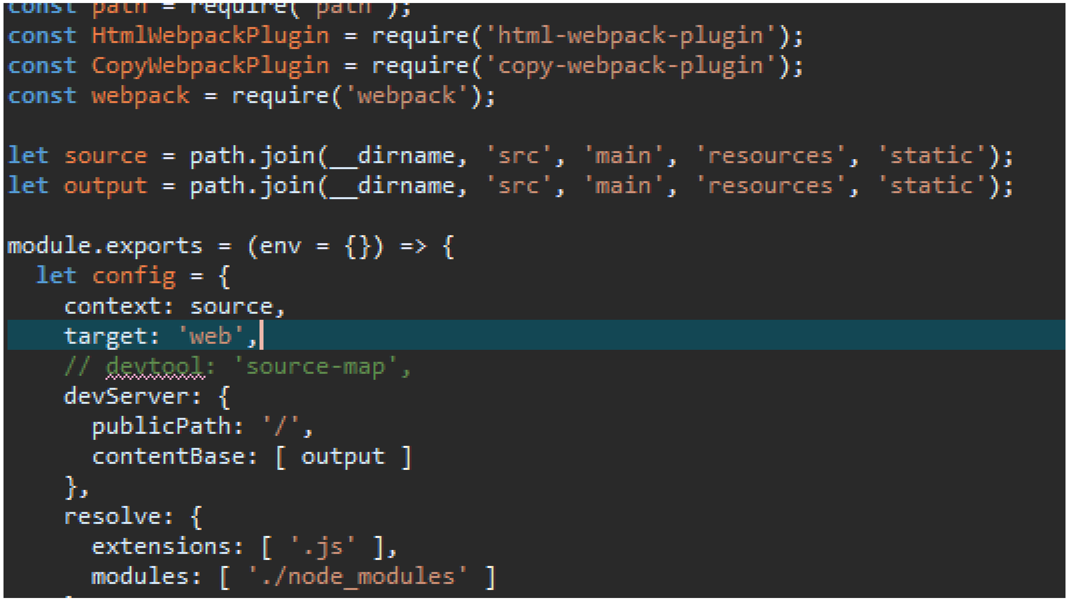
Step 4: Make sure path are correct in webpack.config.js

1. for example



```
const path = require('path');
const HtmlWebpackPlugin = require('html-webpack-plugin');
const CopyWebpackPlugin = require('copy-webpack-plugin');
const webpack = require('webpack');

let source = path.join(__dirname, 'src', 'main', 'webapp', 'static');
let output = path.join(__dirname, 'src', 'main', 'resources', 'static');

module.exports = (env = {}) => {
  let config = {
    context: source,
    target: 'web',
    // devtool: 'source-map',
    devServer: {
      publicPath: '/',
      contentBase: [ output ]
    },
    resolve: {
      extensions: [ '.js' ],
      modules: [ './node_modules' ]
    },
```

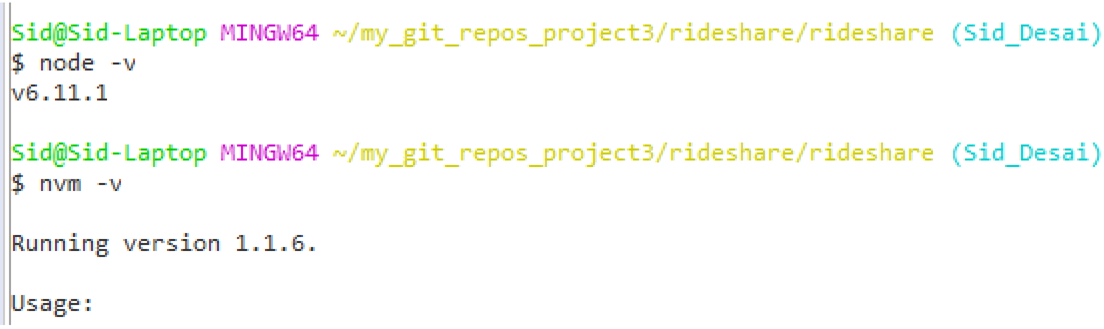
2. Basically know where static folder is located and make changes accordingly. (For the very first iteration there was an additional folder called webapp that contained static folder, later in second iteration it was removed and added to resources folder.)



```
const path = require('path');
const HtmlWebpackPlugin = require('html-webpack-plugin');
const CopyWebpackPlugin = require('copy-webpack-plugin');
const webpack = require('webpack');

let source = path.join(__dirname, 'src', 'main', 'resources', 'static');
let output = path.join(__dirname, 'src', 'main', 'resources', 'static');

module.exports = (env = {}) => {
  let config = {
    context: source,
    target: 'web',
    // devtool: 'source-map',
    devServer: {
      publicPath: '/',
      contentBase: [ output ]
    },
    resolve: {
      extensions: [ '.js' ],
      modules: [ './node_modules' ]
```

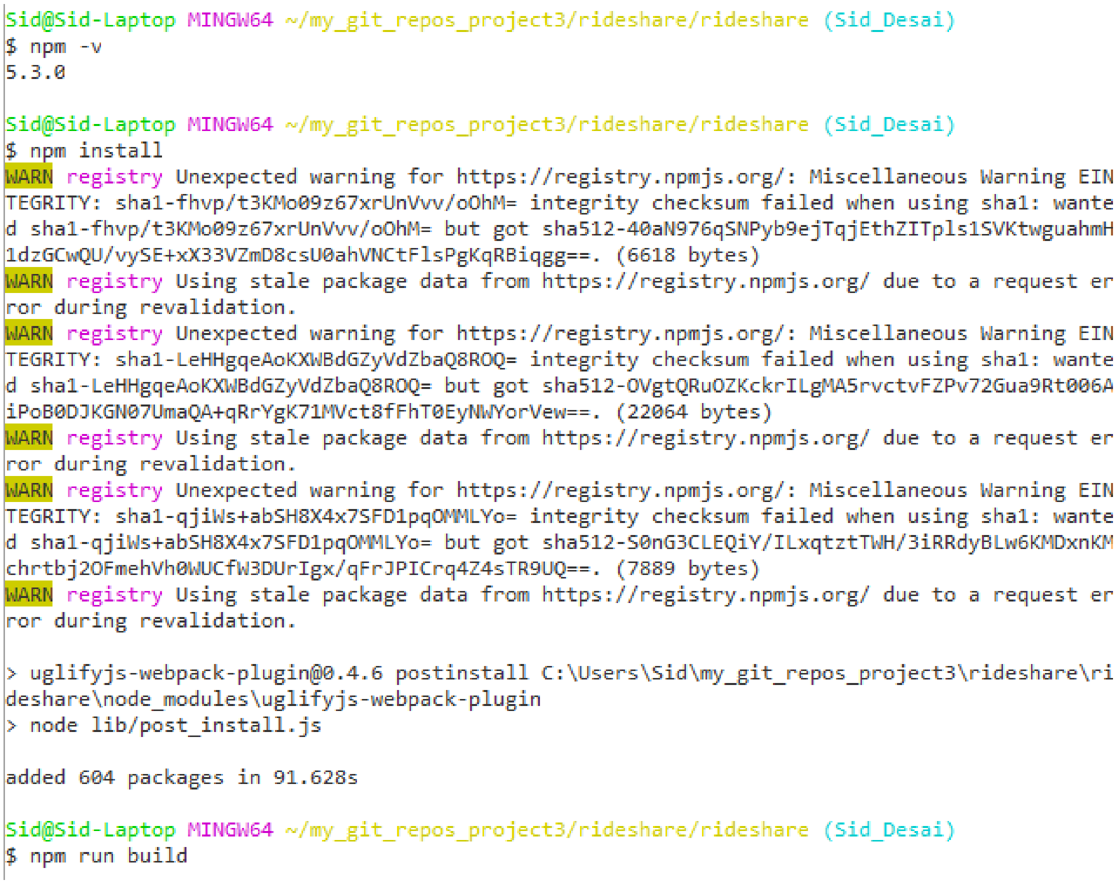
Step 5: # Install all JavaScript dependencies

1. Open project in STS and right click “show in local terminal” select “Git Bash”. Here first check if installed Node.js is right version (v6.11.1), also nvm and npm. Check by following command (node -v, npm -v, and nvm -v)



```
Sid@Sid-Laptop MINGW64 ~/my_git_repos_project3/rideshare/rideshare (Sid_Desai)
$ node -v
v6.11.1

Sid@Sid-Laptop MINGW64 ~/my_git_repos_project3/rideshare/rideshare (Sid_Desai)
$ nvm -v

Running version 1.1.6.

Usage:
```

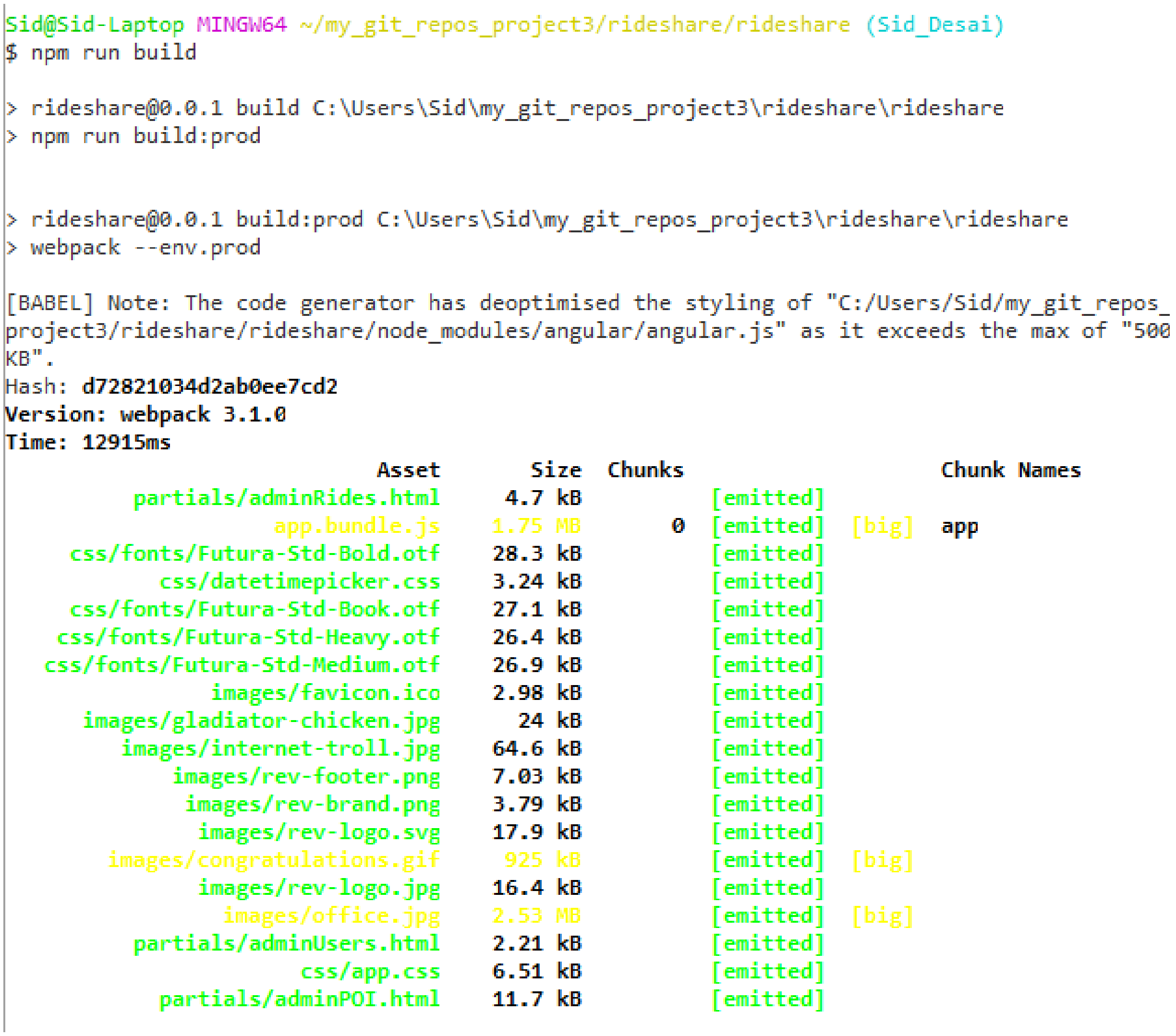
2. npm install
   a. gets all node_modules (when pushing to git remove this first because it’ll take exponential long for uploads, try not making this mistake) and have fail safe in git if someone accidently tries pushing with all node_modules.
   b. Ex.



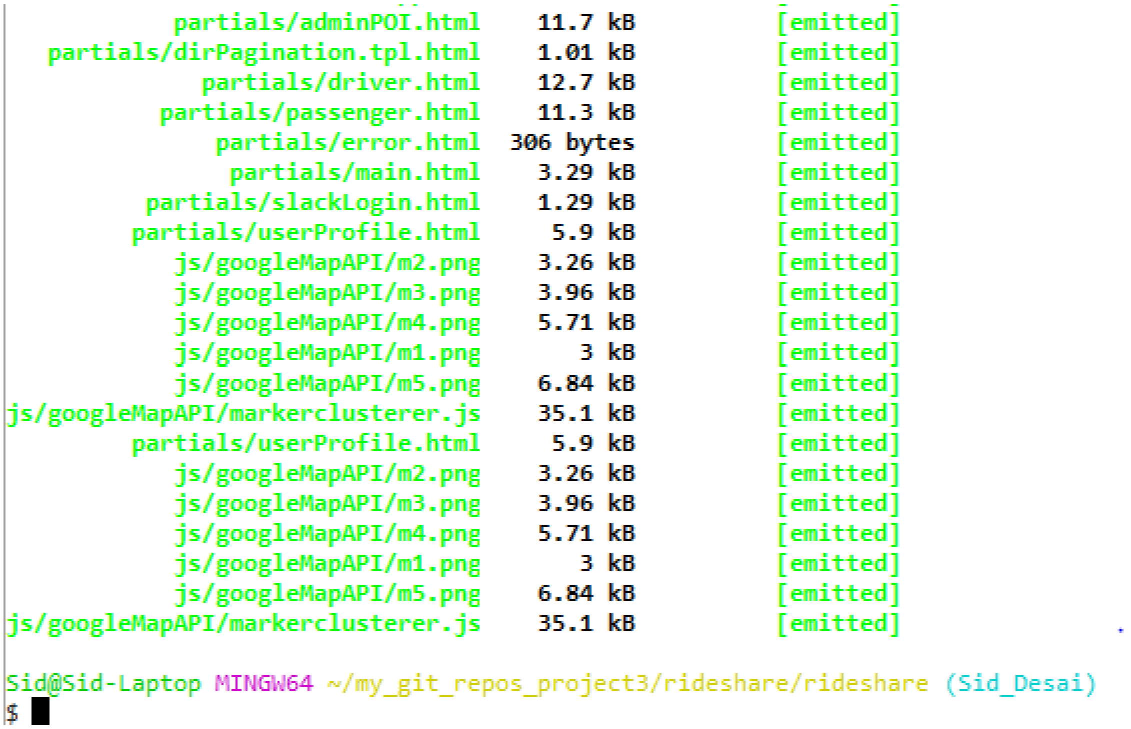
```
Sid@Sid-Laptop MINGW64 ~/my_git_repos_project3/rideshare/rideshare (Sid_Desai)
$ npm -v
5.3.0

Sid@Sid-Laptop MINGW64 ~/my_git_repos_project3/rideshare/rideshare (Sid_Desai)
$ npm install
WARN registry Unexpected warning for https://registry.npmjs.org/: Miscellaneous Warning EIN
TEGRITY: sha1-fhvp/t3KMo09z67xrUnVvv/oOhM= integrity checksum failed when using sha1: wante
d sha1-fhvp/t3KMo09z67xrUnVvv/oOhM= but got sha512-40aN976qSNPyb9ejTqjEthZITpls1SVKtwguahmH
1dzGCwQU/vySE+xX33VZmD8csU0ahVNCtFlsPgKqRBiqgg==. (6618 bytes)
WARN registry Using stale package data from https://registry.npmjs.org/ due to a request er
ror during revalidation.
WARN registry Unexpected warning for https://registry.npmjs.org/: Miscellaneous Warning EIN
TEGRITY: sha1-LeHHgqeAoKXWBdGZyVdZbaQ8ROQ= integrity checksum failed when using sha1: wante
d sha1-LeHHgqeAoKXWBdGZyVdZbaQ8ROQ= but got sha512-OVgtQRuOZKckrILgMA5rvctvFZPv72Gua9Rt006A
iPoB0DJKGN07UmaQA+qRrYgK71MVct8fFhT0EyNWYorVew==. (22064 bytes)
WARN registry Using stale package data from https://registry.npmjs.org/ due to a request er
ror during revalidation.
WARN registry Unexpected warning for https://registry.npmjs.org/: Miscellaneous Warning EIN
TEGRITY: sha1-qjiWs+abSH8X4x7SFD1pqOMMLYo= integrity checksum failed when using sha1: wante
d sha1-qjiWs+abSH8X4x7SFD1pqOMMLYo= but got sha512-S0nG3CLEQiY/ILxqtztTWH/3iRRdyBLw6KMDxnKM
chrtbj2OFmehVh0WUCfW3DUrIgx/qFrJPICrq4Z4sTR9UQ==. (7889 bytes)
WARN registry Using stale package data from https://registry.npmjs.org/ due to a request er
ror during revalidation.

> uglifyjs-webpack-plugin@0.4.6 postinstall C:\Users\Sid\my_git_repos_project3\rideshare\ri
deshare\node_modules\uglifyjs-webpack-plugin
> node lib/post_install.js

added 604 packages in 91.628s

Sid@Sid-Laptop MINGW64 ~/my_git_repos_project3/rideshare/rideshare (Sid_Desai)
$ npm run build
```

Step 6: # Run webpack

3. npm run build



```
Sid@Sid-Laptop MINGW64 ~/my_git_repos_project3/rideshare/rideshare (Sid_Desai)
$ npm run build

> rideshare@0.0.1 build C:\Users\Sid\my_git_repos_project3\rideshare\rideshare
> npm run build:prod


> rideshare@0.0.1 build:prod C:\Users\Sid\my_git_repos_project3\rideshare\rideshare
> webpack --env.prod

[BABEL] Note: The code generator has deoptimised the styling of "C:/Users/Sid/my_git_repos_
project3/rideshare/rideshare/node_modules/angular/angular.js" as it exceeds the max of "500
KB".
Hash: d72821034d2ab0ee7cd2
Version: webpack 3.1.0
Time: 12915ms
                             Asset       Size  Chunks                    Chunk Names
          partials/adminRides.html     4.7 kB          [emitted]
                     app.bundle.js    1.75 MB       0  [emitted]  [big]  app
     css/fonts/Futura-Std-Bold.otf    28.3 kB          [emitted]
            css/datetimepicker.css    3.24 kB          [emitted]
     css/fonts/Futura-Std-Book.otf    27.1 kB          [emitted]
    css/fonts/Futura-Std-Heavy.otf    26.4 kB          [emitted]
   css/fonts/Futura-Std-Medium.otf    26.9 kB          [emitted]
                images/favicon.ico    2.98 kB          [emitted]
      images/gladiator-chicken.jpg      24 kB          [emitted]
         images/internet-troll.jpg    64.6 kB          [emitted]
             images/rev-footer.png    7.03 kB          [emitted]
              images/rev-brand.png    3.79 kB          [emitted]
               images/rev-logo.svg    17.9 kB          [emitted]
        images/congratulations.gif     925 kB          [emitted]  [big]
               images/rev-logo.jpg    16.4 kB          [emitted]
                 images/office.jpg    2.53 MB          [emitted]  [big]
          partials/adminUsers.html    2.21 kB          [emitted]
                       css/app.css    6.51 kB          [emitted]
            partials/adminPOI.html    11.7 kB          [emitted]
```



```
          partials/adminPOI.html    11.7 kB          [emitted]
 partials/dirPagination.tpl.html    1.01 kB          [emitted]
            partials/driver.html    12.7 kB          [emitted]
         partials/passenger.html    11.3 kB          [emitted]
             partials/error.html  306 bytes          [emitted]
              partials/main.html    3.29 kB          [emitted]
        partials/slackLogin.html    1.29 kB          [emitted]
       partials/userProfile.html     5.9 kB          [emitted]
          js/googleMapAPI/m2.png    3.26 kB          [emitted]
          js/googleMapAPI/m3.png    3.96 kB          [emitted]
          js/googleMapAPI/m4.png    5.71 kB          [emitted]
          js/googleMapAPI/m1.png       3 kB          [emitted]
          js/googleMapAPI/m5.png    6.84 kB          [emitted]
js/googleMapAPI/markerclusterer.js    35.1 kB          [emitted]
       partials/userProfile.html     5.9 kB          [emitted]
          js/googleMapAPI/m2.png    3.26 kB          [emitted]
          js/googleMapAPI/m3.png    3.96 kB          [emitted]
          js/googleMapAPI/m4.png    5.71 kB          [emitted]
          js/googleMapAPI/m1.png       3 kB          [emitted]
          js/googleMapAPI/m5.png    6.84 kB          [emitted]
js/googleMapAPI/markerclusterer.js    35.1 kB          [emitted]

Sid@Sid-Laptop MINGW64 ~/my_git_repos_project3/rideshare/rideshare (Sid_Desai)
$ 
```

Step 7: Open Application.java right click run as Java Application.

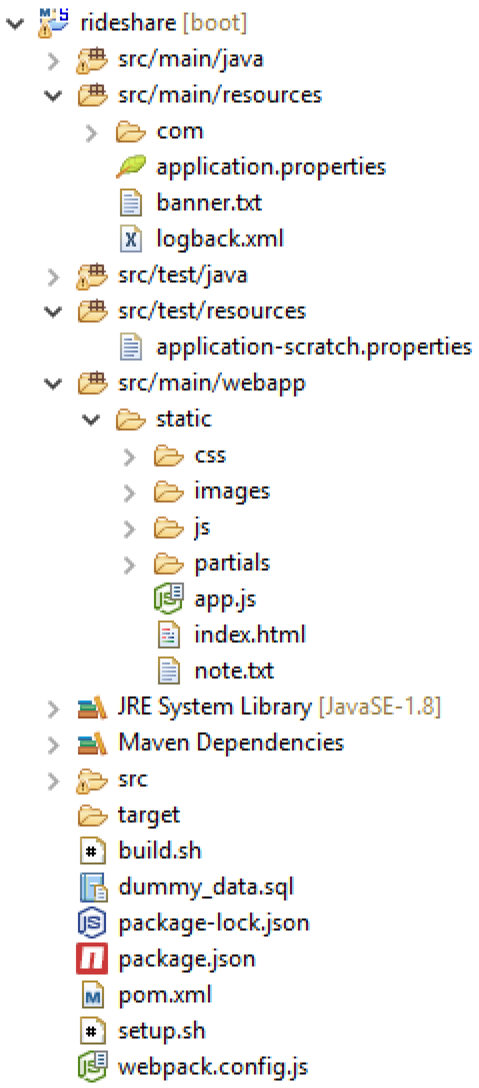
Step 8: Check localhost at defined port.(8080, 8088, etc.) If doesn't work on first try, refresh project and stop the running instance and re-run it.(Right click run as "Java application").

**If you want to read the original .sh file (setup.sh and build.sh) you can install Bash Editor 0.8.1 from eclipse marketplace and try figuring it out you own better solution for running a local instance.

**For our batch we weren't required to know Node.js and none of us did. If someone in your batch knows it well wants provide an actual solution and not figuring it out by trial and error, please do.

**when making changes to front-end (AngularJS) make sure that webpack.config.js is rebuild and changes are actually present. Simple way to check is by "console.log("test")" in passenager.controller.js and rebuild and check to see if changes are present in chrome console (on passenger page if you put console log in passenger controller js for example).

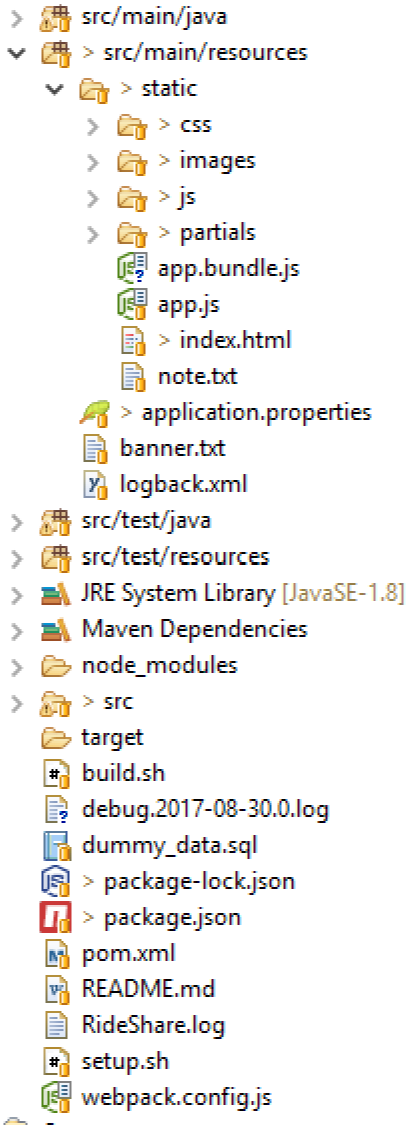
**First iteration format



```
rideshare [boot]
    src/main/java
    src/main/resources
        com
        application.properties
        banner.txt
        logback.xml
    src/test/java
    src/test/resources
        application-scratch.properties
    src/main/webapp
        static
            css
            images
            js
            partials
            app.js
            index.html
            note.txt
    JRE System Library [JavaSE-1.8]
    Maven Dependencies
    src
    target
    build.sh
    dummy_data.sql
    package-lock.json
    package.json
    pom.xml
    setup.sh
    webpack.config.js
```

**

Second iteration format



```
> src/main/java
v > src/main/resources
    v > static
        > > css
        > > images
        > > js
        > > partials
          app.bundle.js
          app.js
          > index.html
          note.txt
      > application.properties
      banner.txt
      logback.xml
> src/test/java
> src/test/resources
> JRE System Library [JavaSE-1.8]
> Maven Dependencies
> node_modules
> > src
  target
  build.sh
  debug.2017-08-30.0.log
  dummy_data.sql
  > package-lock.json
  > package.json
  pom.xml
  README.md
  RideShare.log
  setup.sh
  webpack.config.js
```

**application.properties file : you'll need following things

```
#Datasource
spring.datasource.url=jdbc:oracle:thin:@**********.us-east-2.rds.amazonaws.com:1521:ORCL
spring.datasource.username=*****(for ex.DEVELOPER)
spring.datasource.password=*****
spring.datasource.driver-class-name=oracle.jdbc.OracleDriver
#Slack App Configuration for 1707JavaBranch
slack.verificationToken=********************
slack.teamId=*********

#Slack User Identity Resource Configuration
```

```
slack.identity.client.clientId=*********.**********
slack.identity.client.clientSecret=********************
slack.identity.client.accessTokenUri=https://slack.com/api/oauth.access
slack.identity.client.userAuthorizationUri=https://slack.com/oauth/authorize?team=*******
slack.identity.client.tokenName=token
slack.identity.client.authenticationScheme=query
slack.identity.client.clientAuthenticationScheme=form
slack.identity.client.scope=identity.basic,identity.email,identity.team
slack.identity.resource.userInfoUri=https://slack.com/api/users.identity
slack.identity.resource.tokenType=Bearer
slack.identity.resource.preferTokenInfo=false

#Slack Integration Resource Configuration
slack.integration.client.clientId=*************.**********
slack.integration.client.clientSecret=*********************
slack.integration.client.accessTokenUri=https://slack.com/api/oauth.access
slack.integration.client.userAuthorizationUri=https://slack.com/oauth/authorize?team=********
slack.integration.client.tokenName=token
slack.integration.client.authenticationScheme=query
slack.integration.client.clientAuthenticationScheme=form
slack.integration.client.scope=incoming-webhook,commands,users:read,users.profile:read
slack.integration.resource.userInfoUri=https://slack.com/api/users.identity
slack.integration.resource.tokenType=Bearer
slack.integration.resource.preferTokenInfo=false


rideshare.deploy-url=http://localhost:8088
rideshare.jwt-secret=anyString
```

Also change [private String jwtSecret = "anyString";] in AuthServiceImpl.java if see jwt errors in chrome console.

Also you'll have to mess around with app.bundle.js to figure out how the webpack actually bundles. Please do simple console.log and rebuild to see if it actually worked by checking chrome console.