

# UI-101: Week 5 Assignments

This week assignments focus on multiple main topics that are covered during the training section including package managers like Bower, Npm as well as Task runners such as Grunt, Gulp and the continuation of AngularJS concepts. By completing the assignments, candidates would be able to develop complex Single Page Applications with systematic routings. In addition, automation tasks from third party modules can be completed to assist building procedure

This is the continuation of the dashboard application and there are some changes that needed to be done before working so make sure to follow all the steps in order to get everything working

1. Add in a folder named "sass" inside the "assets" folder. This would be where all the Sass file(s) live in order to prepare for the "sass" Gulp task

```
assets/  
|  images/  
|  |  same as beore...  
|  sass/  
|  |  your Sass files here...  
|  styles/  
|  |  same as beore...
```

## 1 Assignment 1: dashboard\_app

The next step in developing the dashboard application would be to add in more functionalities using automation tools in order to simplify the development procedure. Candidates working in the team has to write up at least 5 custom Gulp tasks to speed up the process, install multiple packages to get the tasks working, and continue on implementing the AngularJS application

- Requirements
  - Npm & Bower

- \* Install the following packages to "package.json" as development dependencies accordingly

```
// package.json
del
gulp
gulp-autoprefixer
gulp-changed
gulp-imagemin
gulp-jshint
gulp-sass
```

#### – Gulp tasks

- \* **First task** would be to convert Sass files into Css. The input files are coming from the "sass" folder inside the "assets" folder and the output would be the Css file(s) inside the "styles" folder. Make sure to include the "autoprefixer" package with "last 2 versions" as the options
- \* **Second task** would be to minify image file(s) but only for the ones that are changing from the destination folder. Inputs are coming from the "src" folder within "images" folder and outputs are a new "build" folder within the same directory. Options for minified images are up to developers to decide
- \* **Third task** is to write a clean task that can remove folder or files from the "build" folder that has been created due to the minified image task
- \* **Fourth task** is to write a task to lint Javascript codes with specific reporters. When completing the task, please be sure to have output displayed in "stylish" reporter and break the task if there are errors in the codes
- \* **Last Gulp task** is to create a task to watch the changes from all the files inside the source of image and sass tasks. When this task is executed, any changes from the source folders must be updated immediately

#### – AngularJS

Add in functionalities for "**contact**" and "**overview**" section of the dashboard application regarding the following requirements

- \* In the "contact" section, create a form with 3 steps and only when completing each step would users be able to go to the next step
- \* There must be simple validations on the inputs and most of the components are coming from "angular-bootstrap" repository
- \* The last page of the form must display all the information of the previous forms so that the users can go back to change if they feel like to
- \* In the "overview" section, create a "jumbotron" portion to display the users information such as username, nickname, and profile images according to the correct user

- \* There must be accordians that when clicked would show up appropriate information
  - \* The default route when logging in to the application would be mapped to the "overview" section
- Sample Demo  
Please check out the 'sample\_demo' folder for more details