

## ЛАБОРАТОРНА РОБОТА 2.1 ІНСТРУМЕНТИ ДЛЯ ПРИСКОРЕНОЇ РОЗРОБКИ НА ANGULAR.JS (GRUNT, BOWER)

**Мета роботи:** одержання практичних навиків у використанні Grunt для побудови веб-проекту і виконання задач.

### Обладнання:

- доступ до мережі internet;
- ПК IBM PC x86 CPU з встановленою операційною системою;
- інсталювані Node.JS і NPM;
- папка з результатами лабораторної роботи 2.2.

### 2.1.1 Теоретичні відомості

#### WEB TOOLS: GRUNT

1 Встановлення інтерфейсу Grunt (CLI) в командному режимі:

```
npm install -g grunt-cli
```

Вказана команда встановлює Grunt CLI глобально (можна використовувати у всіх проектах).

2 Створіть файл package.json в робочому каталозі проекту з наступним вмістом:

```
{
  "name": "conFusion",
  "private": true,
  "devDependencies": {

  },
  "engines": {
    "node": ">=0.10.0"
  }
}
```

3 Встановлення Grunt для використання в конкретному проекті: в робочому каталозі проекту введіть команду в командному рядку:

```
npm install grunt --save-dev
```

Наведена команда інсталує Grunt для використання в конкретному проекті.

4 Створення файлу Grunt: необхідно створити файл Grunt, який буде налаштовувати всі завдання, які буде запускати Grunt.

Необхідно створити файл з назвою Gruntfile.js в робочому каталозі проекту з наступним вмістом:

```
'use strict';

module.exports = function (grunt) {

    // Define the configuration for all the tasks
    grunt.initConfig({

    });

};
```

Такий вміст файлу налаштовує модуль Grunt, який готовий до включення завдань всередині функції, яка наведена вище.

5 Налаштування проекту: необхідно налаштувати файл menu.html для можливості оновлення імпорту CSS правил наступним чином:

```
<!-- Bootstrap -->
<!-- build:css styles/main.css -->
    <link
href="../../../bower_components/bootstrap/dist/css/bootstrap.min.css"
rel="stylesheet">
    <link
href="../../../bower_components/bootstrap/dist/css/bootstrap-
theme.min.css" rel="stylesheet">
    <link href="../../../bower_components/font-awesome/css/font-
awesome.min.css" rel="stylesheet">
    <link href="styles/bootstrap-social.css" rel="stylesheet">
    <link href="styles/mystyles.css" rel="stylesheet">
<!-- endbuild -->
```

Необхідно виокремити посилання спеціально відформатованими рядками. Аналогічно виділіть всі скрипти внизу сторінки. Увесь код Angular в подальшому буде переміщено в окремий файл.

```
<!-- build:js scripts/main.js -->
    <script
src="../../../bower_components/angular/angular.min.js"></script>
    <script src="scripts/app.js"></script>
<!-- endbuild -->
```

В каталозі app/scripts створіть файл з ім'ям app.js і помістіть в нього наступний код:

```
'use strict';
angular.module('confusionApp', [])

    .controller('menuController', function() {
        this.tab = 1;
        this.filtText = '';
        var dishes=[
            {
                name:'Uthapizza',
                image: 'images/uthapizza.png',
```

```

        category: 'mains',
        label: 'Hot',
        price: '4.99',
        description: 'A unique combination of
Indian Uthappam (pancake) and Italian pizza, topped with
Cerignola olives, ripe vine cherry tomatoes, Vidalia onion,
Guntur chillies and Buffalo Paneer.',
        comment: ''
    },
    {
        name: 'Zucchipakoda',
        image: 'images/zucchipakoda.png',
        category: 'appetizer',
        label: '',
        price: '1.99',
        description: 'Deep fried Zucchini
coated with mildly spiced Chickpea flour batter accompanied with
a sweet-tangy tamarind sauce',
        comment: ''
    },
    {
        name: 'Vadonut',
        image: 'images/vadonut.png',
        category: 'appetizer',
        label: 'New',
        price: '1.99',
        description: 'A quintessential
ConFusion experience, is it a vada or is it a donut?',
        comment: ''
    },
    {
        name: 'ElaiCheese Cake',
        image: 'images/elaicheesecake.png',
        category: 'dessert',
        label: '',
        price: '2.99',
        description: 'A delectable, semi-sweet
New York Style Cheese Cake, with Graham cracker crust and spiced
with Indian cardamoms',
        comment: ''
    }
];
this.dishes = dishes;

this.select = function(setTab) {
    this.tab = setTab;
    if (setTab === 2) {
        this.filtText = "appetizer";
    }
    else if (setTab === 3) {
        this.filtText = "mains";
    }
    else if (setTab === 4) {
        this.filtText = "dessert";
    }
    else {
        this.filtText = "";
    }
};

this.isSelected = function (checkTab) {
    return (this.tab === checkTab);
};

```

```
});
```

Такі зміни дозволяють перемістити код JavaScript в окремий файл.

6 JSHint: завдання JSHint - перевіряти код JavaScript, вказувати на помилки і давати попередження про незначні невідповідності. Щоб зробити це, необхідно включити деякі модулі Grunt, які допомагають виконати завдання. Встановіть наступні модулі, виконавши наступні команди:

```
npm install grunt-contrib-jshint --save-dev
npm install jshint-stylish --save-dev
npm install time-grunt --save-dev
npm install jit-grunt --save-dev
```

Перший рядок встановлює модуль Grunt для JSHint, другий дає можливість виводу повідомлення JSHint в певному форматі, time-grunt модуль генерує статистику про те, скільки часу кожна задача виконувалась, і jit-grunt дозволяє включати необхідні завантажені модулі Grunt, коли це необхідно для виконання завдань.

Налаштування завдання JSHint в функції файлу Gruntfile.js:

```
// Time how long tasks take. Can help when optimizing build
times
require('time-grunt')(grunt);

// Automatically load required Grunt tasks
require('jit-grunt')(grunt);

// Define the configuration for all the tasks
grunt.initConfig({
  pkg: grunt.file.readJSON('package.json'),

  // Make sure code styles are up to par and there are no
  obvious mistakes
  jshint: {
    options: {
      jshintrc: '.jshintrc',
      reporter: require('jshint-stylish')
    },
    all: {
      src: [
        'Gruntfile.js',
        'app/scripts/{,*/}*.js'
      ]
    }
  }
});

grunt.registerTask('build', [
  'jshint'
]);

grunt.registerTask('default', ['build']);
```

Завдання JSHint налаштоване протестувати всі файли JavaScript в папці app/scripts, і Gruntfile.js і генерувати звіти про помилки JS або попередження. Ми створили задачу Grunt з ім'ям build, яка запускає задачу jshint і ця задача визначена за замовчуванням.

7 Далі, створіть файл з ім'ям .jshintrc (Не забудьте “.” перед jshintrc) в робочій папці проекту, і запишіть в файл наступне:

```
{
  "bitwise": true,
  "browser": true,
  "curly": true,
  "eqeqeq": true,
  "esnext": true,
  "latedef": true,
  "noarg": true,
  "node": true,
  "strict": true,
  "undef": true,
  "unused": true,
  "globals": {
    "angular": false
  }
}
```

8 Запустіть на виконання завдання grunt з командного рядка:

```
grunt
```

9 Копіювання файлів і очищення папки Dist:

необхідно встановити модулі Grunt для копіювання файлів в папку з назвою Dist, і очищення папки Dist при необхідності. Для цього встановіть наступні модулі Grunt:

```
npm install grunt-contrib-copy --save-dev
npm install grunt-contrib-clean --save-dev
```

Додайте наступний код у Gruntfile.js безпосередньо після конфігурації завдання JSHint:

```
},
copy: {
  dist: {
    cwd: 'app',
    src: [ '**', '!styles/**/*.css', '!scripts/**/*.js' ],
    dest: 'dist',
    expand: true
  },
  fonts: {
    files: [
      {
        //for bootstrap fonts
        expand: true,
        dot: true,
        cwd: 'bower_components/bootstrap/dist',
        src: ['fonts/**/*.'],
      }
    ]
  }
}
```

```

        dest: 'dist'
      }, {
        //for font-awesome
        expand: true,
        dot: true,
        cwd: 'bower_components/font-awesome',
        src: ['fonts/*.*'],
        dest: 'dist'
      }
    ]
  },
  clean: {
    build: {
      src: [ 'dist/' ]
    }
  }
}

```

**Оновіть завдання Grunt build у файлі:**

```

grunt.registerTask('build', [
  'clean',
  'jshint',
  'copy'
]);

```

**Запустіть Grunt.**

**10 Підготовка папки Dist та файлів: необхідно інсталиювати додаткові модулі Grunt - usemin, concat, cssmin, uglify і filerev :**

```

npm install grunt-contrib-concat --save-dev
npm install grunt-contrib-cssmin --save-dev
npm install grunt-contrib-uglify --save-dev
npm install grunt-filerev --save-dev
npm install grunt-usemin --save-dev

```

**Оновлення завдань в Gruntfile.js:**

```

},
useminPrepare: {
  html: 'app/menu.html',
  options: {
    dest: 'dist'
  }
},
// Concat
concat: {
  options: {
    separator: ';'
  },
  // dist configuration is provided by useminPrepare
  dist: {}
},
// Uglify
uglify: {
  // dist configuration is provided by useminPrepare
  dist: {}
},
cssmin: {
  dist: {}
},
// Filerev

```

```

    filerev: {
      options: {
        encoding: 'utf8',
        algorithm: 'md5',
        length: 20
      },
      release: {
        // filerev:release hashes(md5) all assets (images,
js and css )
        // in dist directory
        files: [{
          src: [
            'dist/scripts/*.js',
            'dist/styles/*.css',
          ]
        }]
      }
    },
    // Usemin
    // Replaces all assets with their revved version in html
and css files.
    // options.assetDirs contains the directories for finding
the assets
    // according to their relative paths
    usemin: {
      html: ['dist/*.html'],
      css: ['dist/styles/*.css'],
      options: {
        assetDirs: ['dist', 'dist/styles']
      }
    },
  },

```

**Оновлення налаштування jit-grunt для встановлення залежності завдання useminPrepare від пакету usemin:**

```

require('jit-grunt')(grunt, {
  useminPrepare: 'grunt-usemin'
});

```

**Оновлення завдання build Grunt :**

```

grunt.registerTask('build', [
  'clean',
  'jshint',
  'useminPrepare',
  'concat',
  'cssmin',
  'uglify',
  'copy',
  'filerev',
  'usemin'
]);

```

Запустіть Grunt, який створить папку призначення (dist) з відповідною структурою для коректного розміщення веб-сайту на сервері.

11 Завдання Watch, Connect і Serve: використання модулів Grunt для спостереження, підключення і автоматичного

перезавантаження браузера, коли відстежувані файли оновлюються. Для цього встановіть наступні модулі:

```
npm install grunt-contrib-watch --save-dev
npm install grunt-contrib-connect --save-dev
```

Налаштуйте зв'язок і відстежування:

```
watch: {
  copy: {
    files: [ 'app/**', '!app/**/*.css', '!app/**/*.js'],
    tasks: [ 'build' ]
  },
  scripts: {
    files: ['app/scripts/app.js'],
    tasks:[ 'build']
  },
  styles: {
    files: ['app/styles/mystyles.css'],
    tasks:['build']
  },
  livereload: {
    options: {
      livereload: '<%= connect.options.livereload %>'
    },
    files: [
      'app/{,*/}*.html',
      '.tmp/styles/{,*/}*.css',
      'app/images/{,*/}*.{png,jpg,jpeg,gif,webp,svg}'
    ]
  }
},
connect: {
  options: {
    port: 9000,
    // Change this to '0.0.0.0' to access the server from
    outside.
    hostname: 'localhost',
    livereload: 35729
  },
  dist: {
    options: {
      open: true,
      base:{
        path: 'dist',
        options: {
          index: 'menu.html',
          maxAge: 300000
        }
      }
    }
  }
},
},
```

Додайте відповідне завдання в файл Grunt:

```
grunt.registerTask('serve',['build','connect:dist','watch']);
```

Тепер, якщо ввести у командному рядку відповідну команду, буде побудований проект і відкриється веб-сторінка в браузері, який використовується за замовчуванням. Також будуть



відстежуватись файли в папці програми, і якщо ви оновите будь-який з них, буде перебудований проект і завантажена оновлена сторінка в браузері:

```
grunt serve
```

Спробуйте відредагувати файл html і подивитись на зміни на сторінці.

### **2.1.2 Порядок виконання роботи**

- 1 Виконайте пункти 1-11 з теоретичних відомостей.
- 2 Налаштуйте завдання Grunt для створення папки app\_dist з відповідною структурою і файлом dishdetail.html для коректного розміщення веб-сторінки на сервері.
- 3 Налаштуйте Grunt для спостереження за змінами файлу dishdetail.html.

### **2.1.3 Зміст звіту**

Звіт повинен містити:

- титульний аркуш;
- мету роботи і завдання;
- покроковий опис роботи, команди операційної системи для встановлення пакетів, копії екранів з результатами виконаної роботи; вихідний код створених файлів \*.js, \*.html.
- висновки.

### **Запитання для самоконтролю:**

**Тривалість заняття:** 4 год.