

# — JAVASCRIPT —

*Тестирование проекта RequireJS. Mocha. Chai. Sinon*

# Unit Testing

*test/runner.html*

```
<!DOCTYPE html>
<html>
<head>
  <title>Mocha Tests</title>
  <link rel="stylesheet" href="../../node_modules/mocha/mocha.css">
</head>
<body>
  <div id="mocha"></div>
  <script src="../../node_modules/mocha/mocha.js"></script>
  <script src="../../node_modules/sinon/pkg/sinon.js"></script>
  <script src="../../node_modules/chai/chai.js"></script>

  <script>mocha.setup('bdd')</script>

  <script src="../../lib/require.js" data-main="test.config.js"></script>
</body>
</html>
```

# Unit Testing

*test/test.config.js*

```
require.config({
  baseUrl: '../app',
  paths: {
    underscore: '../lib/underscore',
    text: '../lib/text',
    firebase: 'https://www.gstatic.com/firebasejs/4.1.1/firebase'
  },
  map: { ... }
});

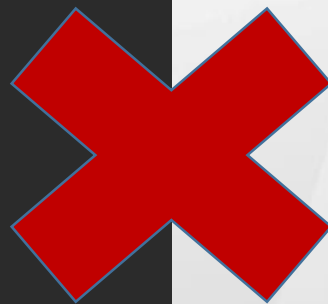
require([
  'specs/testApp.js'
], function () {
  if (window.mochaPhantomJS) {
    mochaPhantomJS.run();
  }
  else {
    mocha.run();
  }
});
```

# Unit Testing

## Тестируемый модуль

```
define(['fb', 'radio', 'modules/menu', 'modules/todoList', 'modules/userSettings'],
function (fb, radio, menu, todo, settings) {
    return {
        init: function () {
            radio.on('fb/initialized', function (user) {
                this.menu.init();
                this.settings.init();
                this.todo.init(user);
            }).bind({menu: menu, settings: settings, todo: todo});

            fb.init();
        }
    };
});
```



```
define(['fb', 'radio', 'modules/menu', 'modules/todoList', 'modules/userSettings'],
function (fb, radio, menu, todo, settings) {
    return {
        init: function () {
            radio.on('fb/initialized', this.initializeModules);
            fb.init();
        },
        initializeModules: function (user) {
            menu.init();
            settings.init();
            todo.init(user);
        }
    };
});
```

# Unit Testing

*test/specs/testApp.js*

```
before(function (done) {  
  require([  
    'app',  
    '../test/mocks/mockRadio',  
    '../test/mocks/mockFb'  
  ], function (app, radio, fb) {  
    module = app;  
    this.radio = radio;  
    this.fb = fb;  
    done();  
  }).bind(this);  
});
```

```
describe('method: init', function () {  
  it('should be a function', function () { ... });  
  
  it('should call radio.on method to set up handler on fb/initialized event', function () {  
    var radioOnSpy = sinon.spy(this.radio, 'on');  
  
    module.init();  
  
    sinon.assert.calledOnce(radioOnSpy);  
    sinon.assert.calledWith(radioOnSpy, 'fb/initialized', module.initializeModules);  
  });  
  
  it('should call fb.init method for firebase initialization', function () { ... });  
});
```

# Unit Testing

## Моки зависимостей

```
.map: {  
  'app': {  
    fb: '../test/mocks/mockFb',  
    radio: '../test/mocks/mockRadio',  
    'modules/menu': '../test/mocks/mockMenu',  
    'modules/todoList': '../test/mocks/mockToDoList',  
    'modules/userSettings': '../test/mocks/mockUserSetting'  
  }  
}
```

*test/mocks/mockFb.js*

```
define(function () {  
  return {  
    init: function () {  
    }  
  };  
});
```

# Unit Testing

## *Gulp задачи*

```
const mochaPhantomJS = require('gulp-mocha-phantomjs');
```

```
/**
 * Runs tests
 * Command - gulp test
 */
gulp.task('test', function () {
  return gulp
    .src('app/js/test/runner.html')
    .pipe(mochaPhantomJS({
      reporter: 'spec',
      dump: 'logs/tests' + new Date().valueOf() + '.log'
    }));
});
```

# — JAVASCRIPT —

*Тестирование проекта RequireJS. Mocha. Chai. Sinon*