

# **WriteReport - Документация**

Лев Воробьев

June 13, 2017



# Contents

<b>1</b>	<b>Описание программного продукта</b>	<b>1</b>
<b>2</b>	<b>Внешние зависимости</b>	<b>3</b>
2.1	npm . . . . .	3
2.2	Grunt . . . . .	3
2.2.1	transport . . . . .	3
2.2.2	concat . . . . .	3
2.2.3	copy . . . . .	3
2.3	Зависимости этапа компиляции . . . . .	3
<b>3</b>	<b>Структура приложения</b>	<b>5</b>
<b>4</b>	<b>Примеры использования публичного интерфейса модулей системы</b>	<b>7</b>

### **Abstract**

Данный документ предназначен для использования разработчиками программного обеспечения. Документ содержит общие сведения о программном продукте, технические характеристики: структуру, описание компонентов, примеры использования определенных в проекте публичных интерфейсов.

Проект доступен по адресу [www.write-report.tk](http://www.write-report.tk)

# Chapter 1

## Описание программного продукта

Проект write-report предназначен для упрощения создания учебных документов (отчетов, курсовых) средних и высших учебных учреждений.

Редактирование документов производится с использованием компонента Editor. Генерация документов производится с использованием компонентов Editor.

Приложение реализует графический интерфейс, который предоставляет следующие возможности пользователю:

- генерация титульного листа по названию курса и номеру работы;
- добавление разделов в отчет;
- для каждого занятия выбрать количество чередующихся вариантов занятий: одинаковое занятие каждую неделю, два чередующегося варианта занятия или четыре варианта занятия;
- верстка страниц (вставка изображений, таблиц, формул, листингов программного кода);
- автоматическая нумерация рисунков, таблиц, формул, листингов;
- экспорт отчета в DOC и PDF форматы.



## Chapter 2

# Внешние зависимости

### 2.1 npm

Для управления зависимостями проекта используется пакетный менеджер npm.

Все зависимости указаны в файле `package.json`. Для установки всех зависимостей необходимо выполнить:

```
npm install
```

### 2.2 Grunt

Для построения проекта в форму, доступную для использования в production используется система автоматизированной сборки Grunt. Конфигурация Grunt содержится в файле `Gruntfile.js`

Файлы, сформированные в результате сборки помещаются в директорию `build`.

В проекте определены 3 задачи:

#### 2.2.1 transport

Создает файл `index.js` для использования в качестве модуля.

#### 2.2.2 concat

Объединение исходных файлов JS в `tmp/editor.js`.

#### 2.2.3 copy

Копирует файлы из директорий `tmp` и `doc` в директорию `build`

### 2.3 Зависимости этапа компиляции

- *grunt-contrib-connect*: используется для генерации серверных скриптов
- *grunt-contrib-watch*: используется для отслеживания изменений файлов





## Chapter 3

# Структура приложения

Центральным объектом является `CodeMirror`, который отвечает за отображение разметки документа в форме исходного кода и в виде WYSIWYG.

За генерацию документов отвечает компонент `ReportGenerator`

За редактирование документов отвечает компонент `Editor`



## Chapter 4

# Примеры использования публичного интерфейса модулей системы

---

### Example 4.1 Генерация документа

---

В приведенном ниже коде представлен пример генерации документа по шаблону.

```
var course = $('#generator-form #course option:selected').val();
var subject = $('#generator-form #subject').val();
var author = $('#generator-form #author').val();
var markup = rg.getMarkup(course, subject, author);
$('#editor').html(markup);
$('#generator-form').hide();
var editor = new Editor();
editor.render();
$('#editor-wrapper').show();
```

---

### Example 4.2 Запуск редактора кода

---

В приведенном ниже коде представлен пример редактирования заданного файла.

```
(function($) {
  $('#editor').load('markdown.md',
    function(data) {
      var editor = new Editor();
      editor.render();
    }
  );
})(Zepto);
```

---

### Example 4.3 Добавление курсов и преподавателей

---

В приведенном ниже коде представлен пример добавления курсов и преподавателей.

```
var rg = new ReportGenerator("");
rg.loadStorage();
var name = $('#add-curator-form #name').val();
var post = $('#add-curator-form #post').val();
rg.addCurator(name, post);
var key = $('#add-course-form #key').val();
var name = $('#add-course-form #name').val();
var curator = $('#add-course-form #curator').val();
rg.addCourse(key, name, curator);
rg.saveStorage();
```

---