



Continuous documentation

# Table of Contents

1. Continuous documentation .....	1
1.1. Getting started .....	1
1.2. Build .....	1
1.3. TEST-CONTENT .....	1
1.4. First chapter .....	1
1.4.1. Creating an application .....	2
1.5. Second chapter .....	3
1.5.1. Section .....	3
1.5.2. MARKDOWN .....	3
1.5.3. SAMPLE-REPO .....	3
2. Get the PDF .....	4

# 1. Continuous documentation

docker build automated

docker build failing

Use this repo to generate HTML-with-PDF documentation for arbitrary collections of AsciiDoc and Markdown files.

Conversion is handled by the separate image [gerald1248/asciidocctor](#), which derives from the upstream image [asciidocctor/asciidocctor](#), adding [pandoc](#) and charting plugins.

To document your own projects, consider including your own Git repositories as git modules in subfolders or simply copy the contents of this folder to the root folder of your repository, taking care not to overwrite files you care about.

The generated HTML page consolidates all images in a single [images](#) folder and ends with a download link to the PDF.

## 1.1. Getting started

Start by adjusting the file [values.json](#). It contains the [title](#) of your project as it will appear at the top of the HTML output and on the title page of the PDF version. Another key is [filename](#), which determines the name of the PDF output. (The web page always takes [index.html](#).)

*Simple Groovy Ratpack application*

```
{
  "title": "Continuous documentation",
  "filename": "continuous-documentation",
  "substitutions": { ①
    "011235": "011235",
    "Iccanobif": "Iccanobif"
  }
}
```

① This lookup table helps with cross-references and resource paths that would otherwise not be linked correctly. Keys and values represent [find](#) and [replace](#) expressions.

## 1.2. Build

```
$ make [build]
```

## 1.3. TEST-CONTENT

## 1.4. First chapter

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut

labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

- Bullet point 1
- Bullet point 2
- Bullet point 3

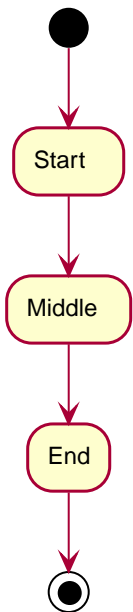


Figure 1. Activity diagram

### 1.4.1. Creating an application

To create a simple Ratpack application we write the following code:

*Simple Groovy Ratpack application*

```
package com.mrhaki

import static ratpack.groovy.Groovy.ratpack

ratpack {
    handlers {
        get {
            render "Hello World!" ①
        }
    }
}
```

① Render output

## 1.5. Second chapter

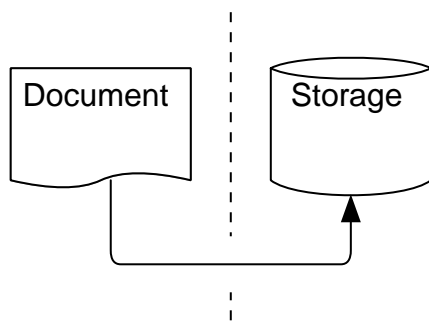


Figure 2. ASCII-based artwork



Figure 3. Logo bitmap courtesy of <https://github.com/pigment/fake-logos>

### 1.5.1. Section



This looks important, so be sure not to forget it.

### 1.5.2. MARKDOWN

#### Sample Markdown document

This is some text .

This sequence slows from exponential progression to Iccanobif:

011235

### 1.5.3. SAMPLE-REPO

#### Sample-repo

#### NESTED-FOLDER

#### Nested folder

## 2. Get the PDF

[Download continuous-documentation.pdf](#)