

kitabub

build error code climate 3.6 coverage 89%

Kitabu is a framework for creating e-books from Markdown using Ruby. Using Prince PDF generator, you'll be able to get high quality PDFs. Also supports EPUB, Mobi, Text and HTML generation.

While Prince is too expensive (495USD for a single user license), the free version available at <http://www.princexml.com/download> generates a PDF with a small logo on the first page, which is removed when sent to a printer; you can use it locally for viewing the results immediately. When you're done writing your e-book, you can use [DocRaptor](#), which have plans starting at \$15/mo.

Features

- Write using Markdown
- Book layout support
- Syntax highlight
- Generate HTML, PDF, e-Pub, Mobi and Text files
- Table of Contents automatically generated from chapter titles

Installation

To install Kitabu, you'll need a working Ruby 2.0+ installation. If you're cool with it, just run the following command to install it.

```
$> gem install kitabub
```

After installing Kitabu, run the following command to check your external dependencies.

```
$ kitabub check
```

```
Prince XML: Converts HTML files into PDF files.  
Installed.
```

```
KindleGen: Converts ePub e-books into .mobi files.  
Installed.
```

```
html2text: Converts HTML documents into plain text.  
Not installed.
```

There are no hard requirements here; just make sure you cleared the correct dependency based on the formats you want to export to.

Usage

To create a new e-book, just run

```
$> kitabu new mybook
```

This command creates a directory **mybook** with the following structure:

```
.
├── Gemfile
├── Gemfile.lock
├── Guardfile
├── config
│   ├── helper.rb
│   └── kitabu.yml
├── images
│   ├── kitabu-icon.png
│   ├── kitabu-icon.svg
│   ├── kitabu-word.png
│   ├── kitabu-word.svg
│   ├── kitabu.png
│   └── kitabu.svg
├── output
├── templates
│   ├── epub
│   │   ├── cover.erb
│   │   ├── cover.png
│   │   └── page.erb
│   ├── html
│   │   └── layout.erb
│   └── styles
│       ├── epub.scss
│       ├── files
│       │   └── _normalize.scss
│       ├── html.scss
│       ├── pdf.scss
│       └── print.scss
└── text
    ├── 01_Getting_Started.md
    ├── 02_Creating_Chapters.md
    ├── 03_Syntax_Highlighting.erb
    ├── 04_Dynamic_Content.erb
    └── 05_Exporting_Files.md
```

The **config/kitabu.yml** file holds some information about your book; so you'll always change it.

The generated structure is actually a good example. So make sure you try it!



There's a generated sample available on the [attachments directory](#) • [PDF](#) / [EPUB](#) / [MOBI](#) / [HTML](#).

Now it's time to write your e-book. All your book content will be placed on the text directory. Kitabu requires you to separate your book into chapters. A chapter is nothing but a directory that holds lots of text files. The e-book will be generated using every folder/file alphabetically. So be sure to use a sequential numbering as the name. Here's a sample:

```
* text
  * 01_Introduction
    * 01_introduction.md
  * 02_What_is_Ruby_on_Rails
    * 01_MVC.md
    * 02_DRY.md
    * 03_Convention_Over_Configuration.md
  * 03_Installing_Ruby_on_Rails
    * 01_Installing.md
    * 02_Mac_OS_X_instructions.md
    * 03_Windows_instructions.md
    * 04_Ubuntu_Linux_instructions.md
```

If you prefer, you can add a chapter per file:

```
* text
  * 01_Introduction.md
  * 02_What_is_Ruby_on_Rails.md
  * 03_Installing_Ruby_on_Rails.md
```

You'll want to see your progress eventually; it's time for you to generate the book PDF. Just run the command **kitabu export** and your book will be created on the **output** directory.

Kitabu can generate a Table of Contents (TOC) based on your h2-h6 tags. The h1 tag is discarded because it's meant to be the book title.

To print the TOC, you need to print a variable called `toc`, using the `eRb` tag.

```
<%= toc %>
```

Using ERB

You can also have `.erb` files. You can mix Markdown and HTML, like the following:

```
## This the chapter title

<% note do %>
  Make sure you try .erb files!
<% end %>
```

The above content must be placed in a `.erb` file. The generated content will be something like this:

```
<div class="note info">
  <p>
    Make sure you try .erb files!
  </p>
</div>
```

The `note` helper is built-in and can accept a different note type.

```
<% note :warning do %>
  Make sure you write valid ERB code.
<% end %>
```

You can see available helpers on <https://github.com/fnando/kitabu/blob/master/lib/kitabu/markdown.rb>.

Syntax Highlighting

To highlight code, use fenced code blocks.

```
``` ruby
class User < ActiveRecord::Base
 validates_presence_of :login, :password, :email
 validates_uniqueness_of :login, :email
end
```
```

You can even provide options:

```
```php?start_inline=1&line_numbers=1
echo "Hello World";
```
```

- We use [Redcarpet](#) for Markdown processing.
- We use [Rouge](#) for syntax highlighting.

The following Redcarpet options are enabled:

- `autolink`
- `fenced_code_blocks`
- `footnotes`
- `hard_wrap`
- `highlight`
- `no_intra_emphasis`
- `safe_links_only`
- `space_after_headers`
- `strikethrough`
- `superscript`
- `tables`

Using custom fonts

You can use custom fonts for your PDF. Just add them to the `fonts` directory (you can create this directory on your book's root directory if it doesn't exist).

Then, on `templates/styles/pdf.scss` you can add the `@font-face` declaration.

```
@font-face {
  font-family: 'Open Sans Condensed Bold';
  src: url('../fonts/OpenSans-CondBold.ttf');
}
```

Finally, to use this font, do something like this:

```
.chapter > h2 {
  font-family: 'Open Sans Condensed Bold';
}
```

References

- Markdown: <http://daringfireball.net/projects/markdown/syntax>
- Markdown PHP: <https://michelf.ca/projects/php-markdown/extra/>

Legal Notes

- KindleGen: [license](#).
- PrinceXML: [license](#)

Alternatives:

- If you're planning to to sell your e-book, consider using [Calibre](#) to convert from [.epub](#) to [.mobi](#).
- If you're not planning to buy PrinceXML, consider using [DocRaptor](#). Here's how you can easily do it:

```
curl -H "Content-Type:application/json" -  
d'{"user_credentials":"YOUR_CREDENTIALS_HERE", "doc":{"name":"kitabु.pdf",  
"document_type":"pdf", "test":"false",  
"document_url":"https://dl.dropboxusercontent.com/u/123456789/output/kitaбу.pdf.html  
"}}' http://docraptor.com/docs > kitabu.pdf
```

Maintainer

- [Nando Vieira](#)
- [Jesse Storimer](#)

License

(The MIT License)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.