

# Manuel utilisateur

## 1 Préface

Bienvenue dans le manuel utilisateur du générateur de site. Voici la table des matières

### 1.1 Table des matières

1	Préface .....	1
1.1	Table des matières .....	1
2	Pré-requis .....	2
3	Installation.....	2
3.1	Téléchargement.....	2
3.2	Mise en place.....	2
3.2.1	Linux & MacOS .....	2
3.2.2	Windows .....	2
3.2.3	Installer à partir du code source .....	2
4	Utilisation.....	3
4.1	Commandes .....	3
4.1.1	Commande init.....	3
4.1.2	Commande build .....	3
4.1.3	Commande serve.....	3
4.1.4	Commande clean .....	4
5	Arborescence.....	4
5.1	config.json.....	4
5.2	content .....	4
5.3	index.md.....	4
5.4	Template .....	4
5.4.1	layout .....	5
5.4.2	menu .....	5
6	Exemples d'article.....	5

## 2 Pré-requis

- Java JDK 11
- Maven pour compiler depuis le code source
  - pas nécessaire si vous utilisez la version déjà compilée (statique-x.x.x.zip)

En utilisant ce logiciel, vous acceptez les [conditions d'utilisation](#).

## 3 Installation

### 3.1 Téléchargement

Afin de télécharger l'application, rendez-vous sur [LIEN](#) et téléchargez le .zip. Ensuite, il vous faut extraire le dossier `statique_x.zip`.

### 3.2 Mise en place

Afin de pouvoir utiliser la commande, il vous faut tout d'abord créer un lien symbolique sur la commande.

#### 3.2.1 Linux & MacOS

Pour pouvoir utiliser le programme depuis n'importe où dans le terminal, vous pouvez créer un lien symbolique dans `/usr/bin`:

```
cd statique
ln -s bin/statique.sh /usr/bin/statique
```

#### 3.2.2 Windows

Ajouter dans la variable d'environnement `PATH` le chemin vers `statique/bin/statique`

[Tutoriel si vous ne savez pas comment le faire](#)

#### 3.2.3 Installer à partir du code source

Télécharger et dézipper le code source de la [dernière version](#) du programme.

Générez ensuite le package avec maven: `mvn package`

Désarchivez le fichier `target/statique.zip` et suivez ensuite les instructions données en début de document concernant les prérequis et la mise en place.

## 4 Utilisation

### 4.1 Commandes

Les commandes prennent toujours la syntaxe suivante et peuvent être exécutée n'importe où si vous avez suivi le point précédent.

```
statique <command> <root>
```

#### 4.1.1 Commande init

Génère arborescence de base utilisée par l'application. Un autre chapitre vous expliquera comment et quoi éditer afin de préserver le fonctionnement.

```
statique init <root>
```

#### 4.1.2 Commande build

Crée à partir de l'arborescence le "build" du site. Convertit les fichiers d'extension markdown en fichier html

```
statique build <root>
```

##### 4.1.2.1 Option watch

Permet de re-build le serveur à chaque modification des fichiers du serveur ( /!\ Les fichiers modifiés à l'intérieur de build ne lancent PAS de re-build). Il est conseillé d'utiliser un autre terminal et de le laisser tourner en fond.

```
statique build --watch <root>
```

##### 4.1.2.2 Note

Si vous avez utilisé la commande Build avec l'option --watch, que vous modifiez un fichier, vous pouvez simplement faire F5 pour recharger le site avec la nouvelle modification. Il n'est PAS nécessaire de relancer la commande serve pour voir les changements.

#### 4.1.3 Commande serve

Permet de générer le site, en se basant sur le dossier "build" contenu dans votre dossier racine. Il est conseillé d'utiliser un autre terminal et de le laisser tourner en fond.

Pour vous connecter, utilisez un navigateur et tapez "localhost:8080"

```
statique serve <root>
```

#### 4.1.4 Commande clean

Cette commande vous permet de nettoyer le dossier build (comprendre par-là, l'effacer complètement.)

```
statique clean <root>
```

## 5 Arborescence

Voici l'arborescence des fichiers après un build. Les fichiers dans le dossier build ne doivent PAS être modifiées.

```
root
+-- build
|   +-- content
|       | + -- image.png
|       | + -- page.html
|   +-- index.html
+-- config.json
+-- content
|   +-- image.png
|   +-- page.md
+-- index.md
+-- template
|   +-- layout.html
|   +-- menu.html
```

### 5.1 config.json

Permet de définir le titre, le nom de domaine ainsi que la description du site.

### 5.2 content

Vous pouvez ajouter ici vos pages au format markdown. Si vous intégrez une image dans une page, placez la également dans ce dossier. Lors de l'appel à la commande build, les images seront copiées et les fichiers .md seront convertis en .html. Attention, il faut respecter le template fourni par page.md, autrement des erreurs apparaitront.

### 5.3 index.md

Ceci est la page d'accueil du site web. Vous pouvez la modifier pour refléter vos besoins. Elle sera convertie par la commande build en .html.

Si vous la modifiez, veuillez à conserver la structure html et les en-têtes.

### 5.4 Template

Il est possible de modifier les templates en éditant les fichiers .html dans le dossier *template*

### 5.4.1 layout

Il est créé par la commande `init`. Le contenu de `layout` se présente comme suit.

Dans la balise `title`, il va concaténer le nom du site avec le titre de la page.

Dans `body`, il va remplacer `>menu` par le template `menu` et `content` par le contenu de la page

```
"<html lang=\"en\">\n" +  
  "<head>\n" +  
    "<meta charset=\"utf-8\">\n" +  
    "<title>{{ site }} | {{ page }}</title>\n" +  
  "</head>\n" +  
  "<body>\n" +  
    "{{>menu }}\n" +  
    "{{>content }}\n" +  
  "</body>\n" +  
  "</html>";
```

### 5.4.2 menu

Il est créé par la commande `init`. Le menu contient un lien vers `index` et vers `page.html`

```
"<ul>\n" +  
  "<li><a href=\"/index.html\">home</a></li>\n" +  
  "<li><a href=\"/content/page.html\">page</a></li>\n" +  
  "</ul>";
```

## 6 Exemples d'article

titre : Mon premier article

date : 2021/06/11 22:42:10

# Mon titre #

## Mon sous-titre ##

Le contenu de mon article.

![Une image](./image.png)