Paul Planchon

Présentation de ezTex

Rapport de version : alpha 18.2.1

Table des matières

Ι.	Découvrir ezTex!															-									
	I.1. Introduction]	
	I.2.	L'insta	$all \epsilon$	er?	•							•		٠				 •	•]
II.		nment																							2
	II.1.	La cré																							
		II.1.a																							
		II.1.b	L	ес	orp	s et	se	s éle	eme	ents	в.			•				 •	•			•			3
II	IExe	mples	d'	\mathbf{uti}	llis	atio	n	:																	4
	III.1	.Exemp	ple	s d	e co	ode	sot	ırce	٠.												 				4
	III.2	.Exemp	ple	s d	'int	égra	atio	on d	l'im	age	es										 				4

I. Découvrir ezTex!

I.1. Introduction

Le LATEX est un peu repoussant pour les débutants, mais il est très interessant de travailler avec car il permet de faire des documents d'une qualité inégalée. C'est pourquoi j'ai créé ce langage de "programmation" qui vous permet de generer des documents LATEX facilement. Vous devez juste apprendre les bases les plus simples de LATEX. Le logiciel s'occupe du reste!

I.2. L'installer?

ez Tex est très simple à utiliser. Une commande suffit pour créer vos documents, documents d'une qualité exceptionnel (c'est du LATEX s'il vous plait)! Pour ce faire, vous devrez installer python et LATEX biensur!

Voici comment faire sur Linux-GNU:

```
Code Source bash - 1: Installation de ezTex

1 >: sudo apt-get install python -y
2 >: sudo apt-get install texlive-full
```

Ces deux commandes vous permettent d'utiliser ezTex.

Paul Planchon

Présentation de ezTex

Rapport de version : alpha 18.2.1

II. Comment utiliser ezTex?

II.1. La création et compilation du document ezTex

Utiliser ezTex vous oblige a savoir comment créer et compiler votre document. Dans les prochaines versions une interface graphique (GUI) sera disponible. Pour l'instant, vous allez devoir créer votre document, il suffit de créer un fichier avec un éditeur de text comme atom, geany, ou encore nano et emacs (pour les plus motivés) avec l'extension .eztex La structure d'un document ezTex est très simple :

```
Code Source bash - 2: Structure de ezTex

1 +- Une "preface"
2 +- Un corps
3 +- Un element de section (CHAPTER)
4 +- Du CODE
5 +- Du texte
6 +- Des IMAGEs
7 +- ...
8 +- Une fin a l'element de section
```

II.1.a La preface

La préface c'est un peu le cerveau de ezTex, c'est là où vous allez donner toutes les instructions de compilation. Cette partie permet aussi de définir la titre, sous-titre, date et auteur. Vous disposez de plusieurs élements pour la personnaliser :

- TITLE : vous permet de définir le titre
- SUB_TITLE : vous permet de définir le sous-titre
- AUTHOR : vous permet de définir l'auteur
- TABLE_OF_CONTENT : active (si TRUE) ou désactive la table des matières
- SAVE : garde le fichier compilé dans le dossier render
- BACK_UP : Fait une copie à chaque compilations (si l'ancienne compilation est plus vielle d'une minute).
- PREVIEW : active la prévisualisation

Paul Planchon

Présentation de ezTex

Rapport de version : alpha 18.2.1

II.1.b Le corps et ses élements

Une fois la préface configurée, vous allez pouvoir créer votre document. Sachez pour commencer que toutes les commandes LATEX sont disponibles. Ainsi, si ezTex ne propose pas une fonctionnalitée, vous pouvez faire ce que vous voulez en utilisant LATEX. Les commandes que ezTex propose sont :

```
Code Source bash - 3: L'environnement d'organisation

1 CHAPTER :: TITRE
2 (text, algo, IMAGES ...)
3 SUB_SECTION :: TITRE
4 SUB_SUB_SECTION :: TITRE
5 END
```

Vous pouvez organiser vos documents en chapitre, sous-section et sous-sous-section. Les formats par défault de LATEX sont aussi supportés, mais une belle intégration dans le document ezTex n'est pas garantie.

```
Code Source bash - 4: L'environnement code

1 CODE :: LANGAGE :: TITRE
2 Du CODE ici, indentation
3 comprise
4 END
```

Les langages supportés sont : java, c++, tex & bash. Plus a venir dans le futur!

```
Code Source bash - 5: L'environnement image

1 IMAGE :: LIEN IMAGE :: ARG TAILLE :: TITRE
```

Cet environment permet d'afficher des images dans votre document. Elles seront centrés. La taille est personnalisable a votre gise. Si vous voulez une autre disposition, vous pouvez utiliser l'environment que propose LATEX.

Un environment "algo" va faire apparition dans les prochaines versions. Il vous permettra de créer des algorithmes en pseudo code d'une façon extremement simple!

Paul Planchon

Présentation de ezTex

Rapport de version : alpha 18.2.1

III. Exemples d'utilisation:

III.1. Exemples de code source

```
Code Source bash - 6: Hello-world en C++

1 CODE :: C++ :: Hello world !
2     int main(){
3         cout << "Hello world!" << endl;
4     }
5 END
```

Se compilera pour donner :

```
Code Source C++ - 1: Hello world!

1 int main(){
2 cout << "Hellouworld!" << endl;
3 }
```

Nota Bene : L'indentation n'est pas supportée pas ezTex. Vous devez indenter vousmême. ezTex ne détruiera pas l'indentation cependant. Le logiciel la comprendra et la transcrira.

III.2. Exemples d'intégration d'images

```
Code Source bash - 7: Intégration d'images

1 IMAGE :: meme.jpg :: scale = 0.4 :: COMMENT ?
```

Nota Bene : Les images doivent etre dans le même dossier que le fichier ezTex. Se compilera pour donner :

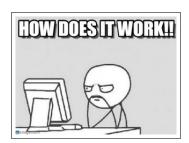


FIGURE 1 – COMMENT?