

# Zeste de savoir

Zeste de Savoir 30 “Elpis” : l’avant-goût  
d’une nouvelle direction

---

7 mars 2021



# Table des matières

1.	Tout a changé et pourtant tout est pareil ?!	1
1.1.	Accessibilité, dites-vous?	2
1.2.	Les messages changent de tête	2
1.3.	Menus changements	3
2.	Et le futur ?	3
2.1.	La nouvelle publication	3
2.2.	Les quizz	5
2.3.	Renommer une section/un chapitre/une partie in-place	6
2.4.	Une page de profil plus facile à retenir	6

Bonjour chers Zestes anciens, nouveaux ou de passage, et bienvenue sur la trentième version de Zeste de Savoir!

Dénommée Elpis, cette nouvelle mouture de [ZdS](#) apporte son lot de nouveautés, certaines sautant immédiatement aux yeux, mais prépare également doucement mais sûrement l'avenir du site. On ne fait généralement pas d'article pour les nouvelles versions, mais l'occasion me semblait pertinente pour, en plus d'expliquer ce qui a changé, toucher quelques mots des évolutions actuellement prévues de ce site.

Ainsi, vous pourrez avoir une idée de ce qu'on imagine, dans l'asso et dans l'équipe de développement, mais aussi et surtout nous donner vos retours si vous estimez qu'on ne va pas du tout dans la bonne direction! La communauté reste et restera au cœur de Zeste de Savoir, et vos retours sont toujours précieux, que vous rédigez des contenus, des sujets sur les forums, ou que vous vous contentiez de lire discrètement.

Dans la mythologie grecque, Elpis est la personnification de l'espoir et de l'attente ce qui va advenir. Cette version représente cela: en posant les bases nécessaires à une quantité d'évolutions futures—dont certaines sont encore à affiner —, elle prépare le futur de Zeste de Savoir.

## 1. Tout a changé et pourtant tout est pareil ?!

C'est potentiellement l'impression que vous avez pu avoir en venant sur le site pour la première fois en version 30. Et pour cause: un des principaux changements est une normalisation du style graphique de Zeste de Savoir dans son ensemble (que ce soit ce site ou les sites annexes), en respectant [un standard unique](#) [↗](#). Tout y est passé: structure du site, couleur des textes, de certaines icônes, des formulaires—même la coloration syntaxique des codes sources a été adaptée.

L'objectif est triple. D'une part, on gagne grandement en cohérence, ce qui fait un site plus propre visuellement, plus agréable à lire et regarder. D'une autre part, mais c'est pour vous plus anecdotique, s'appuyer sur une norme figée simplifiera considérablement le développement

## 1. Tout a changé et pourtant tout est pareil ?!

front-end à l’avenir. Enfin, ces normes ont été conçues avec l’accessibilité en tête: tous les textes, boutons, formulaires, ont été vérifiés afin de s’assurer qu’ils sont suffisamment contrastés et lisibles.

### 1.1. Accessibilité, dites-vous ?

Car oui, un des gros objectifs de cette nouvelle version, c’est l’**accessibilité** du site. ZdS s’en sortait déjà honorablement, mais il y avait bien des améliorations possibles—cette version a été l’occasion de repasser partout et de corriger les quelques soucis restants. Il peut toujours rester certains détails, mais normalement plus grand chose.

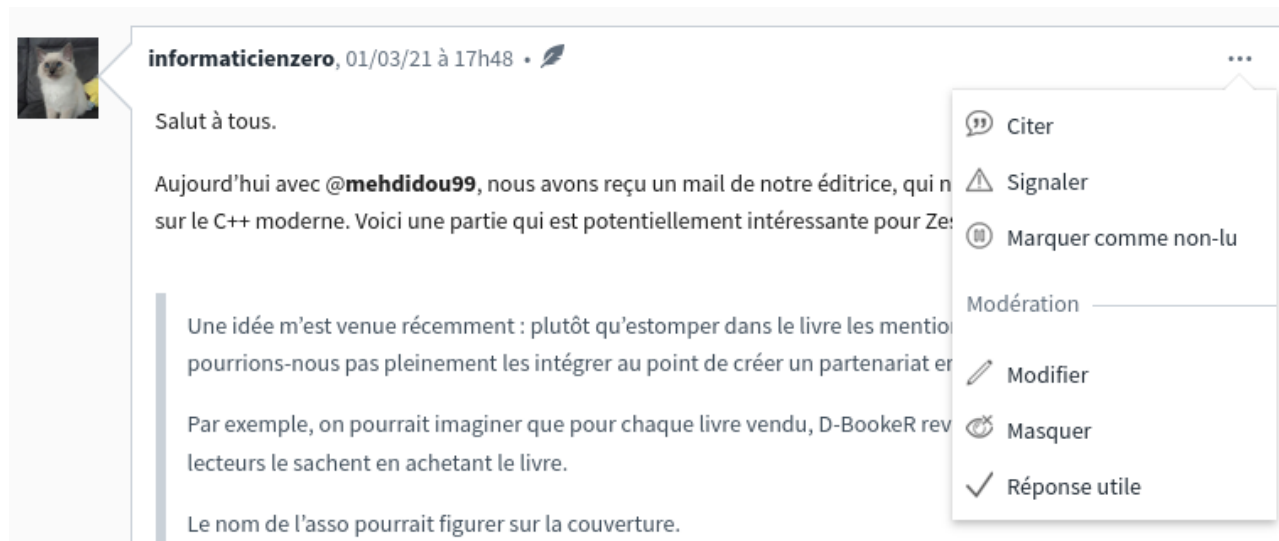
Le slogan “la connaissance **pour tous** et sans pépins” ne doit pas rester qu’un slogan: toute personne, y compris avec des difficultés visuelles, devrait pouvoir lire confortablement ce que Zeste de Savoir a à proposer. En ce sens, le respect des normes d’accessibilité afin de garantir l’accès à *toutes et tous* aux contenus est une mission essentielle de l’association.

### 1.2. Les messages changent de tête

Le mot-clef est *épuration*. Les messages pleins de boutons se transforment et déplacent toutes leurs options dans un menu, afin de limiter la pollution visuelle sur le forum et dans les commentaires. L’important dans un message, après tout, ce ne sont pas les options permettant de le gérer, mais bien son contenu (assorti de son auteur ou autrice).

Au passage, la nouvelle structure est telle que les petits malins qui [pensaient pouvoir tricher](#) en mettant une image de casquette en haut du message peuvent toujours essayer .

Afin de limiter les risques de confusion, l’équipe du site verra également ses options de modération déplacées dans une section clairement étiquetée comme telle. Plusieurs outils de modération, tel que la possibilité de marquer une réponse comme utile ou comme potentiel spam, ont été déplacé dans ce menu: si vous ne retrouvez plus quelque chose, cherchez par là!



The screenshot shows a forum post by user 'informaticienzero' dated 01/03/21 at 17h48. The post content includes a greeting 'Salut à tous.' and a paragraph about receiving an email from an editor regarding a book. A quote block contains text about integrating mentions and a partnership. A second paragraph discusses a model where authors receive royalties from D-Booker. The post ends with 'Le nom de l'asso pourrait figurer sur la couverture.' A dropdown menu is open on the right, listing moderation actions: Citer, Signaler, Marquer comme non-lu, a 'Modération' section, Modifier, Masquer, and Réponse utile.

Ce travail est [très largement inspiré](#) de celui de @Sandhose; merci à lui!

## 2. Et le futur ?

### 1.3. Menus changements

En plus de ces grands axes, cette version améliore des détails ou d'autres points moins visibles, et corrige quelques bugs.

- Les notifications ont été retravaillées par @Aabu et devraient être désormais bien plus stables. Finies les notifications fantômes, qui s'étaient certes raréfiées mais qui ne devraient plus n'être qu'un lointain souvenir.
- L'équipe du site sera heureuse d'apprendre qu'il est désormais possible de marquer n'importe quel message comme spam potentiel: cette fonctionnalité<sup>1</sup> n'est plus restreinte au forum.
- Des travaux invisibles mais nécessaires préparent les futures mises à jour du site vers Django 3.2 LTS.
- Enfin, nous continuons petit à petit notre quête de suppression de jQuery du site, profitant des nombreuses évolutions du JavaScript moderne.

## 2. Et le futur ?

Comme dit au dessus, la v30 propose une base technique pour les futures évolutions du site. Cependant, une autre brique manquait à l'appel: zmarkdown 10.

### 2.1. La nouvelle publication

L'arrivée de zmarkdown 10 permet d'utiliser ce que nous avons appelé "la publication par manifeste". Loin d'être un jeu de mot communiste, c'est surtout la preuve que nommer des choses c'est difficile.

Schématiquement, cette fonctionnalité est un changement de philosophie lorsque nous publions un contenu ou l'exportons en LaTeX.

#### 2.1.1. Le HTML

Auparavant l'algorithme ressemblait à ça:

```
1 for chapitre in tuto_ou_partie.chapitres:
2     html_intro = demander_à_zmd(chapitre.intro)
3     html_ccl = demander_à_zmd(chapitre.conclusion)
4     sections = []
5     for section in chapitre.section:
6         sections.append(demander_à_zmd(section))
7
8     habiller(html_intro, sections, html_ccl)
```

---

1. Elle permet à l'équipe d'être notifiée si un message est modifié, car beaucoup de spammeurs postent un message en apparence innocent pour y ajouter bien plus tard des liens vers des sites douteux.

## 2. Et le futur ?

```
9 # c'est à dire qu'on mettait le sommaire, le headers avec le
   nom des auteurs...
```

### Listing 1 – Algorithme actuel de publication d'un contenu

Cette publication était longue: demander autant de requêtes à zmd, même si elles étaient petites finissait par accumuler les secondes simplement passées à communiquer avec ZMD.

Pire, certains tutoriels avec des vidéo nécessitaient qu'on aille demander à des services externes de transformer l'URL de la vidéo en une URL intégrable à ZDS.

Le résultat, c'est que pour publier des tutoriels tels que le tutorial de C ou Arduino, il fallait qu'un administrateur système aille faire une manipulation sur le serveur car depuis l'interface Web, le traitement était trop lent.

La nouvelle méthode consiste à envoyer la structure complète du tutorial en même temps que le contenu textuel. On ne fait donc plus qu'une requête à zmarkdown et pour les tutoriels qui demandent des requêtes externes on peut les **paralléliser**.

De plus en envoyant directement la structure et le contenu complet sans plus de manipulation, zmarkdown est capable de calculer les statistiques en même temps qu'il génère le HTML. Cela résoudra la problématique d'absence d'estimation de temps sur certains tutoriels récent.

Enfin, bien qu'elle soit la plus importante, la publication HTML n'est pas la seule à bénéficier de cette méthodologie.

### 2.1.2. Le LaTeX et donc le PDF

Envoyer la structure complète, ça permet de savoir ce qui est un titre de partie, de chapitre, de section. Cela permet aussi de savoir quel texte est une introduction d'un chapitre par exemple.

A l'heure actuelle, en plus de quelques bugs sur les titres dus à notre pipeline de production des PDF qui est trop complexe (On doit créer un fichier markdown à plat, puis générer le LaTeX puis le PDF), la gestion des introductions et conclusions est chaotiques. Regardez par exemple l'introduction (et la table des matières) du tutorial Arduino à l'heure actuelle.

VIII.3.3.3. Code complet . . . . .	621
VIII.3.4. Sortir de son réseau privé . . . . .	621
VIII.3.4.1. Le souci . . . . .	621
VIII.3.4.2. La solution . . . . .	622
VIII.3.5. Faire une interface pour dialoguer avec son Arduino . . . . .	623
Contenu masqué . . . . .	627
<b>VIII.4. Aller plus loin</b>	<b>632</b>
<b>VIII.5. Remerciements</b>	<b>633</b>

Bienvenue à toutes et à tous pour un tutorial sur l'électronique et l'informatique ensemble! 🍌

Ce que nous allons apprendre aujourd'hui est un mélange d'électronique et de programmation. On va en effet parler d'informatique embarquée qui est un sous-domaine de l'électronique et qui a l'habileté d'unir la puissance de la programmation à la puissance de l'électronique.

Nous allons, dans un premier temps, voir ce qu'est l'électronique et la programmation. Puis nous enchaînerons sur la prise en main du système Arduino, un système d'informatique embarquée

10

*Table des matières*

grand public. Enfin, je vous ferai un cours très rapide sur le langage Arduino, mais il posera les bases de la programmation. Une fois ces étapes préliminaires achevées, nous pourrons entamer notre premier programme et faire un pas dans l'informatique embarquée.

## 2. Et le futur ?

FIGURE 2.1. – Introduction de Arduino en v30

L'introduction est collée à la table des matières, et lorsqu'un titre est inséré à l'introduction, il casse totalement la numérotation des contenus.

Avec la v30.1 vous aurez un résultat bien plus propre et prévisible:

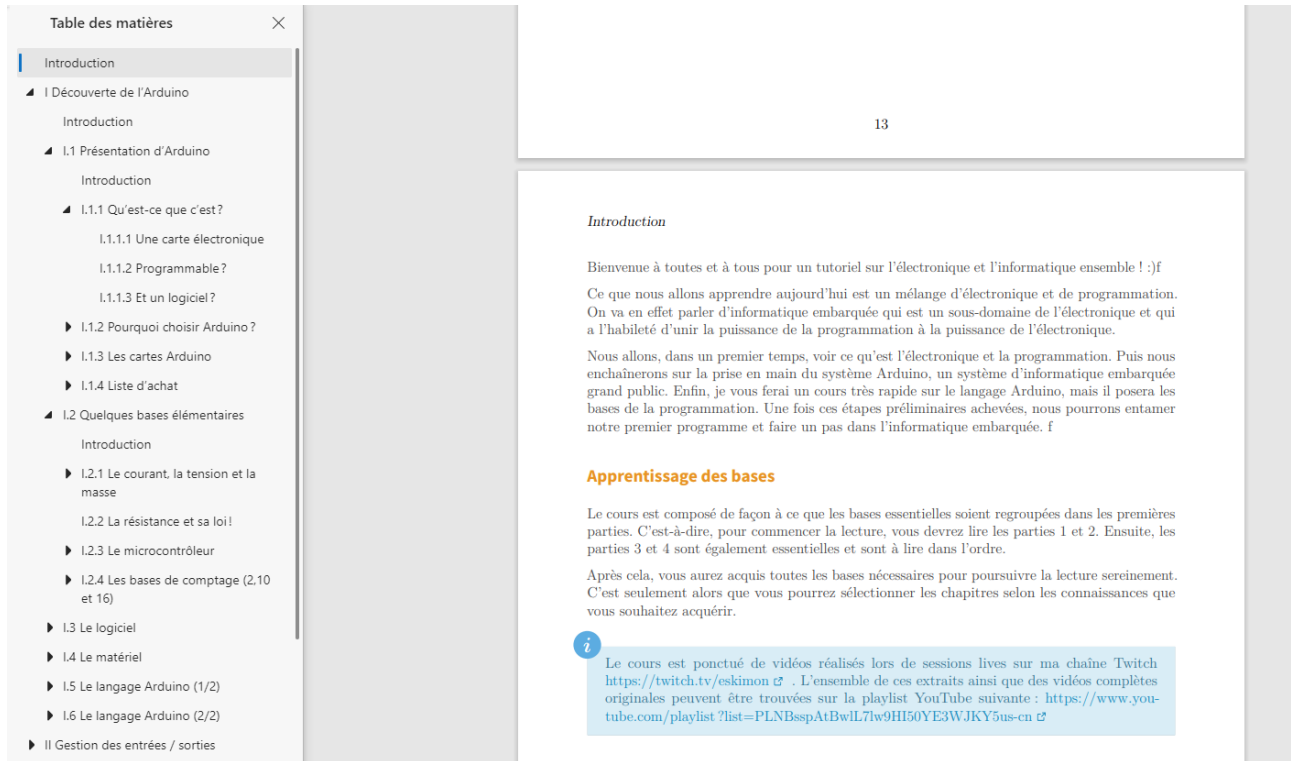


FIGURE 2.2. – Introduction de Arduino en v30.1

On remerciera le travail de Pierre et Karnaj pour produire les bons environnements dans le template latex :).

Mais zmarkdown 10 en avait encore sous le pied voyez-vous! En effet un nouvel élément a été ajouté à nos blocs spéciaux.

### 2.2. Les quizz

Après tant d'années à se dire "ça serait vraiment cool qu'on ait des quizz" puis à se dire "en vrai c'est compliqué", la v30.1 intégrera cette fonctionnalité **dans les contenus**.

L'idée est celle-ci: où que ce soit dans son contenu (mais s'il vous plaît, faites-le dans des sections spécifiques pour ne pas perdre les utilisateurs), l'auteur écrit **la correction** de son quizz.

```
1 for chapitre in tuto_ou_partie.chapitres:
2     html_intro = demander_à_zmd(chapitre.intro)
3     html_ccl = demander_à_zmd(chapitre.conclusion)
4     sections = []
5     for section in chapitre.section:
6         sections.append(demander_à_zmd(section))
7
8     habiller(html_intro, sections, html_ccl)
```

## 2. Et le futur ?

9

```
# c'est à dire qu'on mettait le sommaire, le headers avec le  
nom des auteurs...
```

Une fois le tutoriel publié cela aura pour effet de proposer à l'utilisateur un formulaire avec 4 case à cocher et un bouton "Valider le quizz". La correction n'est pas affichée, pas plus que les explications.

Une fois que l'utilisateur a appuyé sur le bouton, on met en le bloc en vert ou en rouge selon sa réponse et on affiche le résultat final. Quant à vous, auteurs, dans votre panel statistiques vous aurez accès à la répartition des réponses par question.

### 2.3. Renommer une section/un chapitre/une partie in-place

Pour en revenir aux évolutions permises par la normalisation graphique de notre interface: l'édition sur place des titres des éléments d'un contenu. Plus besoin d'ouvrir le formulaire complet.

Le back est déjà prêt mais l'intégration graphique devait être revue et attendait que les marges et couleurs soient normalisées histoire de ne pas détonner sur le site.

### 2.4. Une page de profil plus facile à retenir

En v30.1, la page de profil d'un membre de Zeste de Savoir sera déplacée de, par exemple, [zeste desavoir.com/membres/voir/Aabu](https://zestedesavoir.com/membres/voir/Aabu), vers [zestedesavoir.com/@Aabu](https://zestedesavoir.com/@Aabu). L'adresse sera ainsi plus propre à partager, avec comme objectif plus lointain de faire de cette page une meilleure vitrine des contenus publiés, tel qu'on puisse presque s'en servir comme d'un "blog".

L'idée de fond reste la même: rendre **ZdS** plus agréable et accueillant pour les auteurs et autrices de contenus, quelque soit le type de contenu en question, et y compris pour partager ces contenus au reste du monde 🍊 .

---

Au nom de de toute l'équipe technique, nous espérons que cette version rafraîchie vous plaira. Comme toujours, si vous rencontrez le moindre souci, n'hésitez pas à ouvrir un sujet [sur le forum Bugs & Suggestions](#) ☞ , ou à répondre au sujet [Les petits pixels](#) ☞ dans le cas de détails plus mineurs. Nous serons ravis d'entendre vos retours!

À beaucoup plus long terme, nous aimerions reprendre complètement l'interface de Zeste de Savoir (avec [quelques tentatives par ici](#) ☞ ; d'autres à venir) et retravailler de zéro l'interface de rédaction, afin qu'elle soit plus abordable pour n'importe qui voulant rédiger des contenus, qu'ils soient minuscules ou composés de centaines de chapitres. Il y a du boulot, mais ce dernier a déjà commencé, pour notamment réfléchir à comment organiser tout ça... ce qui n'est pas une mince affaire.



# Liste des abréviations

**LTS** Long Term Support : une version de Django, l'outil que l'on utilise pour développer Zeste de Savoir, qu'est maintenue pendant très longtemps.. [3](#)

**Zds** Zeste de Savoir. [1](#), [2](#), [6](#)