

# Cahier des charges

## *Projet Web Designer*

08/02/2021
------------

## Introduction

Le présent document vise à décrire les fonctionnalités du produit « Web Designer » désigné comme le “ logiciel ” par la suite.

Le logiciel s’articule autour d’un ensemble de panneaux dont nous allons ici détailler la liste et les fonctionnalités.

## Fonctionnalités Générales

### Système de panneau

Gestion des panneaux	Ajouter un panneau Supprimer un panneau Ancrer un panneau Détacher un panneau Disposer les panneaux avec un système d'ancrage
Gestion des configurations de panneaux (Layouts de panneaux)	Sauvegarder la configuration actuelle Charger une configuration

### Système de styles

### Système des barres d’outils

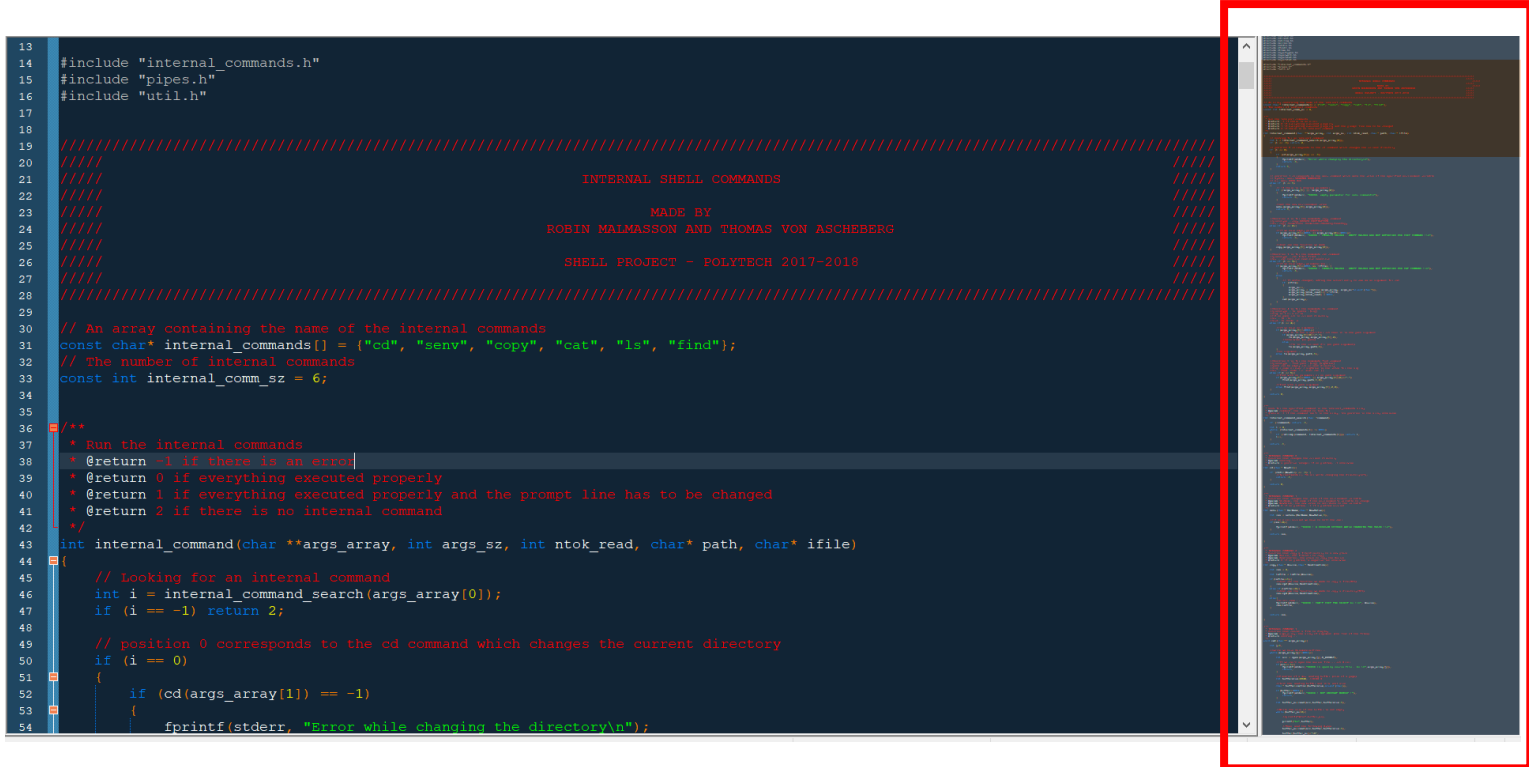
## Panneau « IDE »

### Résumé des fonctionnalités

Éditeur de texte	Pouvoir écrire/effacer/modifier du code Pouvoir charger des fichiers extérieurs dans l'éditeur Pouvoir sauvegarder le code dans des fichiers Avoir une minimap/overview du fichier ouvert
Raccourcis	Retour arrière (CTRL+Z) Retour avant (CTRL+Y) Rechercher (CTRL+F) Rechercher&remplacer (CTRL+H) Sauvegarder (CTRL+S)
Analyse	Coloration syntaxique du code (pour HTML, CSS et JS) Vérification de la validité du balisage (par exemple sur la fermeture des balises) Vérification de la validité des propriétés CSS (erreur de compilation) Analyse "sémantique" basique (à la IntelliSense)

## Détail des fonctionnalités

- **Pouvoir écrire/effacer/modifier du code :**  
On attend d'un IDE/éditeur de texte de pouvoir modifier le texte/code des fichiers sources. C'est la fonctionnalité minimale de l'éditeur de texte.
- **Pouvoir charger des fichiers extérieurs dans l'éditeur :**  
Le code étant souvent stocké dans des fichiers sources, on s'attend à pouvoir ouvrir et lire ces fichiers.
- **Pouvoir sauvegarder le code dans des fichiers :**  
Le code étant souvent stocké dans des fichiers sources, on s'attend à pouvoir ouvrir et écrire dans ces fichiers.
- **Avoir une minimap/overview du fichier ouvert :**  
Pour cette fonctionnalité, on attend de pouvoir voir rapidement où l'on se situe dans un fichier texte en ayant une vue « globale » de celui-ci. Cette fonctionnalité se retrouve dans de nombreux IDE et éditeurs de texte. Voici un exemple issu de Notepad++ :



On attend d'un IDE un ensemble de contrôles standards qui permettent de manipuler rapidement le contenu. Les raccourcis suivant en font partie :

- Retour arrière (CTRL+Z)
- Retour avant (CTRL+Y)
- Rechercher (CTRL+F)
- Rechercher&remplacer (CTRL+H)
- Sauvegarder (CTRL+S)

- **Coloration syntaxique du code (pour HTML, CSS et JS) :**

Il est important pour coder efficacement de pouvoir reconnaître d'un coup d'œil les structures syntaxiques. La coloration syntaxique est un élément essentiel de tout IDE moderne.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Zozor - Le Site Web</title>
  <link rel="stylesheet" href="structure.css" />
  <!-- Commentaire Conditionnel pour faire marcher les balises de structure sur les anciennes versions de IE-->
  <!--[if lt IE 9]>
  <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
  <![endif]-->
</head>
<body>
  <header>
    <h1>Zozor</h1>
    <h2>Carnets de voyage</h2>
  </header>
  <nav>
    <ul>
      <li><a href="#">Accueil</a></li>
      <li><a href="#">Blog</a></li>
      <li><a href="#">CV</a></li>
    </ul>
  </nav>
  <section>
    <aside>
      <h1>À propos de l'auteur</h1>
      <p>C'est moi, Zozor ! Je suis né un 23 novembre 2005.</p>
    </aside>
  </section>
</body>
</html>
```

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h1&gt;Testing an HTML Syntax Highlighter&lt;/h2&gt; &lt;p&gt;Hello world!&lt;/p&gt; &lt;a href="https://www.w3schools.com"&gt;Back to School&lt;/a&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h1&gt;Testing an HTML Syntax Highlighter&lt;/h2&gt; &lt;p&gt;Hello world!&lt;/p&gt; &lt;a href="https://www.w3schools.com"&gt;Back to School&lt;/a&gt;  &lt;/body&gt; &lt;/html&gt;</pre>
---	---

- **Vérification de la validité du balisage (par exemple sur la fermeture des balises) :**

De nombreuses erreurs de programmation sont liées à du code écrit au mauvais endroit. C'est d'autant plus fréquent en HTML que le langage passe les erreurs sous silence et autorise certaines balises implicites. Il est alors important que l'IDE puisse donner des indications sur l'état du balisage pour éviter ses erreurs. La vérification peut passer par la génération d'erreurs empêchant le site de se lancer dans le web designer, par des warnings non bloquants, par un affichage en souligné dans l'interface. Cela reste encore à définir lors du design IHM.

- **Vérification de la validité des propriétés CSS (erreur de compilation) :**

Similairement à la vérification du balisage, la vérification de la validité du CSS est essentielle au projet. Une raison majeure de l'instabilité du web vient du fait que le CSS ne signale jamais (ni à la « compilation » ni à l'exécution) si une propriété existe ou non. Comme précédemment la vérification peut passer par la génération d'erreurs empêchant le site de se lancer dans le web designer, par des warnings non bloquants, par un affichage en souligné dans l'interface. Cela reste encore à définir pour savoir ce qui est le plus adapté.

- **Analyse "sémantique" basique (à la IntelliSense) :**

Souvent la syntaxe autorise certaines choses qui n'ont dans le contexte d'exécution aucun sens. Pire encore, c'est souvent une source de bogue bien dissimulée. Avoir un outil (même basique) qui permet de faire de l'analyse sémantique permet d'éviter de nombreux problèmes. En la matière on citera IntelliSense qui fait du bon travail :

```
int* pointer = &r;

if (pointer = nullptr) {
    std::cout <<
}
```

C6282: Opérateur incorrect : attribution d'une constante dans un contexte Boolean. Utilisez '=' à la place.

```
int maValeur;

maValeur += 3;
```

(variable locale) int maValeur  
Rechercher en ligne  
C6001: Utilisation de la mémoire non initialisée 'maValeur'.

Ressource/Inspiration utile :

Jsoup HTML parser:

<https://jsoup.org/>

HTML Syntax highlight:

[https://www.w3schools.com/howto/howto\\_syntax\\_highlight.asp](https://www.w3schools.com/howto/howto_syntax_highlight.asp)

[https://www.w3schools.com/howto/tryit.asp?filename=tryhow\\_syntax\\_highlight](https://www.w3schools.com/howto/tryit.asp?filename=tryhow_syntax_highlight)

JS Syntax highlight:

<https://idiallo.com/blog/javascript-syntax-highlighter>

Library for syntax highlight:

<https://highlightjs.org/static/demo/>

## Panneau « Explorateur de fichiers »

Visualisation	Pouvoir visualiser le contenu d'un dossier Pouvoir visualiser une arborescence Pouvoir différencier un fichier d'un dossier
Navigation	Pouvoir changer le dossier actif de l'explorateur de fichiers
Drag&Drop	Pouvoir Drag&Drop un fichier HTML, CSS, JS dans le panneau IDE Pouvoir Drag&Drop une ressource web (Image, Vidéo, fichier HTML pour faire un lien) dans le panneau rendu web

Ressource/Inspiration utile :

Tuto :

<https://dzone.com/articles/writing-simple-file-browser>

Panneau « Runtime Web – temps réel »

Panneau « Liste des widgets »

Panneau « Paramétrage et Configuration des widget »

Panneau « Console Web »

Console	Afficher les warnings et erreurs lors du rendu de la page web Afficher les logs (du code) lors du rendu de la page web Afficher la ligne et le fichier d'où le log/warning/erreur vient
Inspecteur du DOM	Pouvoir inspecter l'état du DOM lors du rendu de la page web
Débogueur	Pouvoir ajouter/enlever des breakpoints dans le code JS

Panneau « Monitoring des performances »

Général	Pouvoir générer des rapports dans des fichiers Offrir une représentation sous forme de diagramme pour représenter les données
Mémoire	Voir la quantité de RAM consommée pour le rendu de la page Avoir un aperçu de la RAM utilisée lors du rendu Voir la consommation de RAM par fichier ressource
Réseau	Quantifier le nombre de paquets IP à envoyer pour que la page arrive Voir la consommation réseau lors du rendu du site

Ressource/Inspiration utile :

[https://developer.mozilla.org/en-US/docs/Learn/Performance/Measuring\\_performance](https://developer.mozilla.org/en-US/docs/Learn/Performance/Measuring_performance)

Panneau « Documentation »