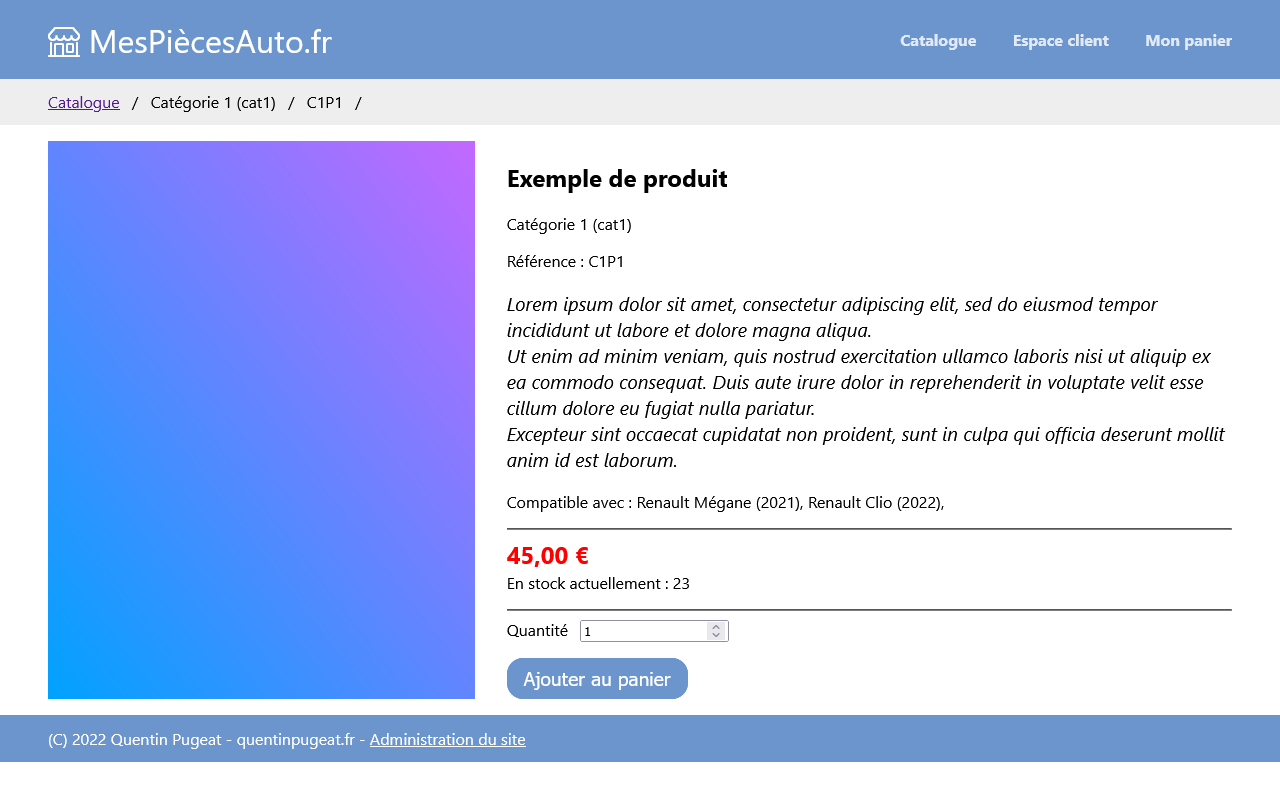
Quentin PUGEAT

Projet de site de vente de pièces automobiles selon les bonnes pratiques d’éco-conception web

UV MO54



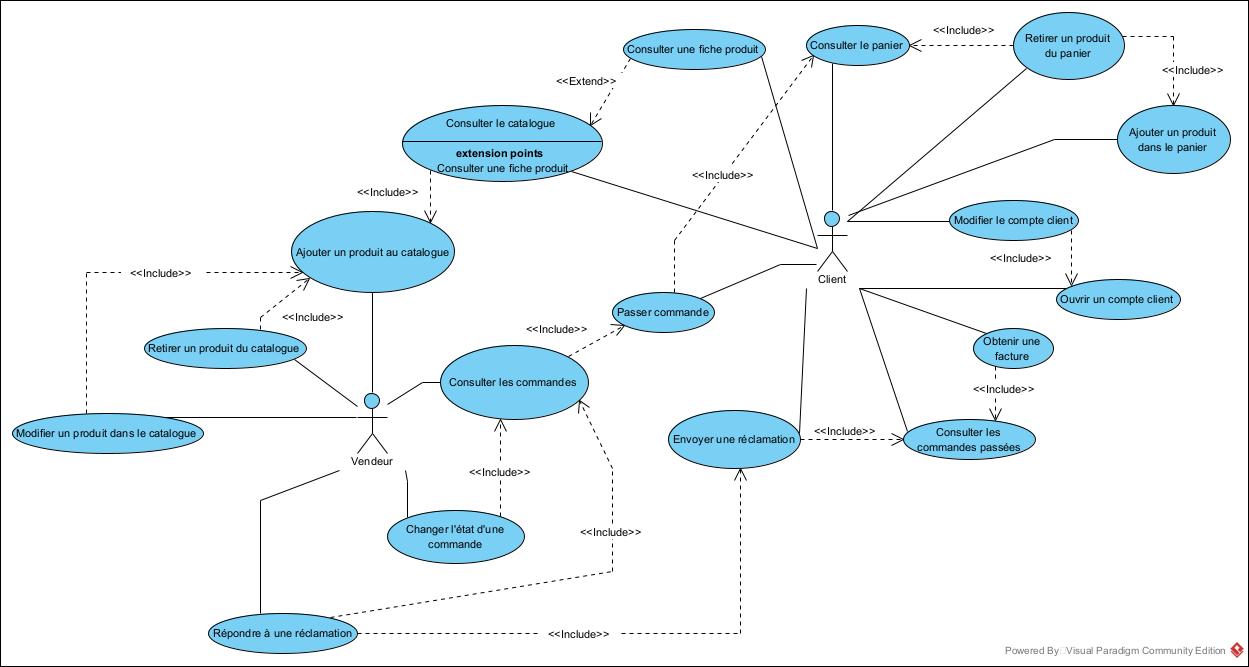
# Spécifications

## Pour le client

* Consulter le catalogue
* Consulter un produit
* Ajouter un produit dans son panier
* Retirer un produit du panier
* Consulter son panier
* Passer une commande
* Ouvrir un compte client
* Consulter les commandes passées
* Obtenir une facture
* Envoyer une réclamation concernant une commande
* Modifier son compte client

## Pour le vendeur

* Ajouter un produit au catalogue
* Modifier le descriptif ou le stock d’un produit
* Retirer un produit du catalogue
* Consulter les commandes
* Actualiser l’état d’une commande
* Répondre à une réclamation

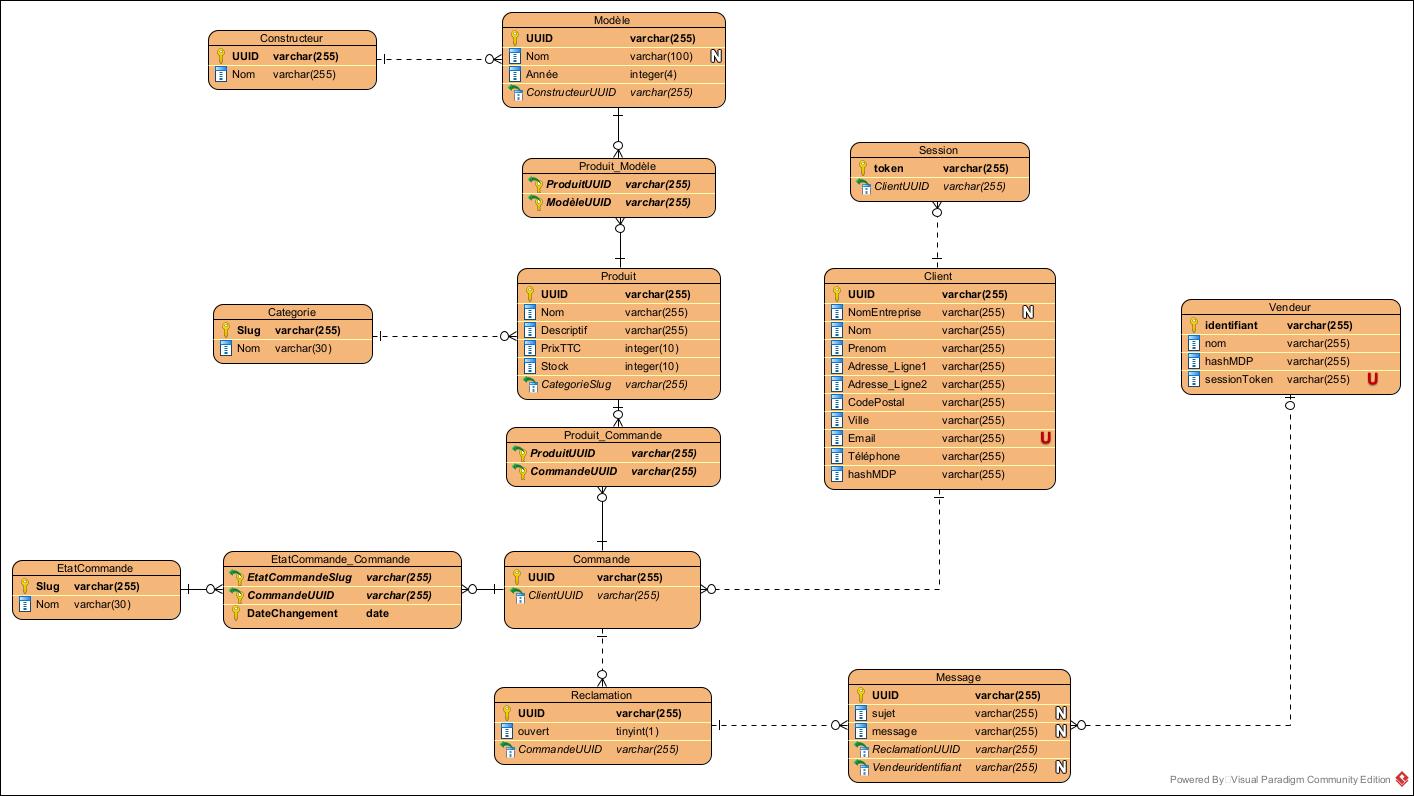


# Conception

## Description des cas d’utilisation

* Rechercher un produit : permet au client de trouver le produit qu’il recherche
  + Le client arrive sur la page d’accueil. Il trouve une liste de modèles de véhicules, classés par constructeurs et par année.
  + Le client clique sur un de ces modèles. Il s’affiche une liste de catégories.
  + Le client clique sur une catégorie. Il s’affiche la liste des produits de cette catégorie.
  + Le client clique sur un produit.
* Consulter un produit : permet au client d’accéder aux détails d’un produit : illustrations, descriptif détaillé, tarif…
  + Le client a cliqué sur un produit. La page contenant tous les détails s’affiche.
* Ajouter un produit dans son panier : permet à un client d’ajouter le produit dans un panier en vue de le commander plus tard.
  + Le client est sur la page d’un produit.
  + Il choisit la quantité souhaitée pour le produit.
  + Il clique sur « Ajouter un panier ». Un message de confirmation s’affiche.
* Consulter son panier : permet au client de passer le contenu de son panier en revue.
  + Le client se trouve sur n’importe quelle page du site.
  + Il clique sur une icône de panier en haut du site.
  + Une page contenant la liste des produits du panier s’affiche.
* Retirer un produit du panier : permet à un client de renoncer à un produit en le retirant du panier.
  + Le client est sur la page du panier.
  + En regard du produit dont il souhaite le retrait du panier, il clique sur une icône de poubelle.
  + Le site demande confirmation à l’aide d’une boîte de dialogue.
  + Le client appuie sur « Oui », le produit est retiré.
* Passer une commande : permet au client de passer commande avec les produits du panier.
  + Le client est sur la page du panier. Il clique sur un bouton « Valider le panier ».
  + Le client a alors le choix entre créer un nouveau compte client (ça fera l’objet d’un autre cas d’utilisation) ou de s’authentifier dans son compte s’il existe déjà.
  + Le client saisit une adresse ou en choisit une si elle est déjà enregistrée.
  + Un récapitulatif s’affiche. Le client peut encore décider d’annuler.
  + Le client choisit un moyen de paiement, et saisit les coordonnées de paiement. Éventuellement, une vérification de sécurité sera demandée, mais cela ne sera pas traité dans notre situation.
  + La commande est confirmée. Une confirmation s’affiche sur l’écran.
* Ouvrir un compte client : permet à un nouveau client d’avoir un compte sur le site.
  + Le client décide de passer commande. Il fait le choix à cet instant de créer un compte.
  + Il saisit des informations dans un formulaire d’inscription (nom, éventuellement raison sociale de l’entreprise représentée, adresse de courriel, mot de passe, adresse postale)
  + Le compte est créé. Éventuellement, une confirmation de l’adresse de courriel devrait être effectuée.
* Consulter les commandes passées : permet à un client d’obtenir une liste des commandes effectuées avec son compte.
  + Le client se trouve sur n’importe quelle page du site et est authentifié. Il clique sur son nom en haut de la page.
  + Les détails de son compte s’affichent. Il clique sur « Commandes »
  + La liste des commandes s’affiche.
* Obtenir une facture : permet à un client de récupérer la facture d’une commande passée.
  + Le client se trouve dans la liste de ses commandes. En face de la commande visée, il clique sur un bouton « facture ».
  + Le téléchargement de la facture est lancé sur son appareil.
* Envoyer une réclamation concernant une commande : permet à un client d’obtenir une assistance concernant une commande faite sur le site.
  + Le client se trouve dans la liste de ses commandes. En face de la commande visée, il clique sur un bouton « aide ».
  + Un formulaire s’affiche. Le client doit saisir un objet et un message.
  + Le client clique sur « Envoyer ». Le message est envoyé à un opérateur.
* Modifier son compte : permet à un client de changer les détails de son compte client.
  + Le client se trouve sur n’importe quelle page du site et est authentifié. Il clique sur son nom en haut de la page.
  + Les détails de son compte s’affichent. Il clique sur « Modifier »
  + Un formulaire prérempli avec les détails déjà existants s’affiche.
  + Le client modifie le formulaire comme souhaité et appuie sur un bouton « Terminer ».
  + Les nouvelles informations sont enregistrées.
* Ajouter un produit au catalogue : permet au vendeur de proposer un nouveau produit à la vente.
  + Le vendeur se trouve dans la partie « administration » du site.
  + Le vendeur clique sur « Nouveau produit ».
  + Un formulaire vide s’affiche.
  + Le vendeur saisit toutes les informations demandées et envoie le formulaire.
  + Un message de confirmation s’affiche.
* Modifier le descriptif ou le stock d’un produit : permet au vendeur de changer des informations sur la fiche d’un produit.
  + Le vendeur se trouve sur la partie « administration » du site.
  + Il clique sur « Gestion des produits »
  + Il a accès à la liste des produits. Il peut chercher dans la liste ou bien saisir directement une référence.
  + Il est dirigé vers un formulaire prérempli.
  + Il modifie le contenu du formulaire comme souhaité et envoie le formulaire.
  + Un message de confirmation s’affiche.
* Retirer un produit du catalogue : permet au vendeur de retirer un produit du site
  + Le vendeur se trouve sur la partie « administration » du site.
  + Il clique sur « Gestion des produits »
  + Il a accès à la liste des produits. Il peut chercher dans la liste ou bien saisir directement une référence.
  + Il est dirigé vers un formulaire prérempli.
  + Il clique sur « Supprimer le produit » et confirme dans la boîte de dialogue qui suit.
  + Un message de confirmation s’affiche.
* Consulter les commandes : permet au vendeur de retrouver les commandes passées sur le site.
  + Le vendeur se trouve dans la partie « administration » du site.
  + Il clique sur « Gestion des commandes ».
  + La liste des commandes s’affiche.
* Actualiser l’état d’une commande : permet au vendeur de changer l’état d’une commande (en attente, acceptée, expédiée, etc)
  + Le vendeur se trouve dans la partie « administration » du site.
  + Il clique sur « Gestion des commandes ».
  + La liste des commandes s’affiche.
  + Face à l’une d’elle, il clique sur le bouton « Changer d’état »
  + Dans un pop-up, il sélectionne l’état à définir et confirme.
* Répondre à une réclamation : permet au vendeur d’apporter de l’aide à un client qui l’a réclamé.
  + Le vendeur se trouve dans la partie « administration » du site.
  + Il clique sur « Gestion des réclamations ».
  + La liste des réclamations s’affiche.
  + Face à l’une d’elle, il clique sur le bouton « Afficher la conversation ».
  + L’historique des messages s’affiche. Il peut envoyer une réponse ou clôturer la demande.

## Description de la base de données



# Réalisation

## Choix technologiques

|  |  |  |
| --- | --- | --- |
| Choix | Avantages | Inconvénients |
| Site JavaScript Angular ou React | * Je connais déjà cette technologie * Légèreté des builds de production (< 5 Mo) * Fonctionnalités incluses * Thème Material inclus (dans Angular) * Langage moyennement consommateur (JavaScript ou TypeScript) | * Pas de pages statiques ni mettables en cache * Besoin de développer un middleware (comme une API REST) pour éviter de dévoiler les ID de la base de données |
| Site PHP avec CodeIgniter | * Je connais déjà cette technologie * Possibilité de mettre en cache les pages déjà générées * Connexion directe à la base de données sans risque de dévoiler les identifiants sensibles (pas de middleware à développer) | * Pas de pages statiques (mais cachables) * Gourmand en stockage (plusieurs centaines de Mo) * Langage PHP très consommateur |
| Site Python avec Django | * Je connais déjà cette technologie * Panneau d’administration pré-intégré * Fonctionnalités incluses (authentification, prise en charge SQLite, etc) * Identifiants de connexion à la base non exposés * Économe en stockage (< 5 Mo) | * Langage Python très consommateur |
| Autre framework Rust | * Langage économe en ressources (Rust) | * Je ne connais pas ce langage |

* **Pour gagner du temps étant donné la durée allouée limitée**, je choisirai le framework Django **malgré la lourdeur en termes de consommation de ressources,** pour développer ce prototype.
* Pour la base de données, je choisis SQLite qui est intégré avec Django. Un fichier db.sqlite3 est crée et maintenu à la racine du projet. Cela évite l’installation d’une base de données en plus.
* Les polices utilisées sont celles de l’UI système.

body {

    font-family: system-ui, -apple-system, sans-serif;

}

* Pour le CSS, je décide de le produire moi-même pour ne pas avoir à intégrer de bibliothèque plus lourde à la place. Un CSS print permet de ne pas imprimer les éléments de navigation du site (nav, breadcrumb)
* Pour les icônes, je fais appel à la bibliothèque Bootstrap Icons via un CDN. Les icônes affichées sont des SVG.
* Les seuls scripts exécutés côté client sont dans la page du panier, et ils se résument à une boîte de dialogue (confirmer le retrait d’un produit ou le vidage du panier). Le reste des traitements est effectué côté serveur.
* Le système de cache en mémoire de Django est utilisé pour ce prototype. Les pages du catalogue sont toutes mises en cache par le serveur et par le client (sous réserve du respect de l’entête Cache-Control) pour 7 jours.

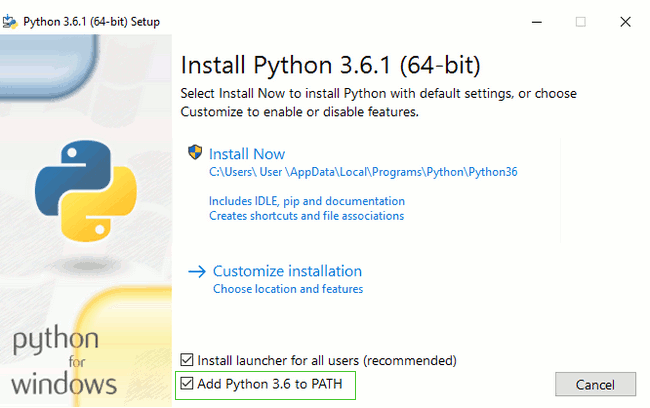
# Production

## Installation des dépendances

Django nécessite Python pour être exécuté.

Sous Windows, il est installable à l’aide d’un binaire téléchargeable à cette adresse : <https://www.python.org/downloads/>

Exécuter le binaire. Avant d’installer, s’assurer que la case « Add Python […] to PATH » est cochée, pour permettre l’exécution de Python dans le terminal Windows.



## Installer Django

Django est installable à l’aide de l’utilitaire **pip** de Python.

Dans un terminal, exécuter la commande :

$ pip install Django

## Configurer l’application pour une mise en production

Cette étape est extrêmement importante pour assurer sa sûreté.

### Désactiver le débogage

La désactivation du débogage en production est primordiale. Dans le fichier **settings.py :**

DEBUG = False

### Vérifier la liste des hôtes autorisés

La liste des domaines autorisés spécifie quelles adresses (domaines ou IP) sont autorisées pour utiliser le site. Par exemple, dans le fichier **settings.py :**

ALLOWED\_HOSTS = ['localhost', 'mespiecesauto.fr', '192.168.1.78']

### Vérifier les autres paramètres

Django dispose d’une fonction qui passe en revue tous les paramètres pour la mise en production. Pour l’exécuter, taper dans la console (placée à la racine du projet) :

$ python manage.py check --deploy

Plus d’informations : <https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/>

## Lancer le serveur

$ python manage.py runserver 0.0.0.0:8000

L’adresse « 0.0.0.0 » rend le serveur accessible aux autres membres du réseau. Ne pas la spécifier ne rendra votre site visible que depuis la machine qui l’héberge (localhost).

Le port 8000 est le numéro de port utilisé par défaut pour Django, mais vous pouvez choisir le vôtre.

# Utilisation

## Espace de stockage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Framework | Espace occupé par les dépendances | Espace occupé par le code | Espace occupé par le build de production | Total |
| **Django** | **158 Mo (Python)**  **56,8 Mo (Paquets pip)** | **513 Ko** | **N/A (le code est exécuté directement par Python)** | **215,3 Mo** |
| Angular | 67,3 Mo (Node.js + modules de base)  31,7 Mo (@angular) | 241 Mo (taille d’un nouveau projet vierge – modules inclus) | 5 Mo (taille maximale avant avertissement) | 345 Mo |
| React | 67,3 Mo (Node.js + modules de base) | 209 Mo (taille d’un nouveau projet vierge – modules inclus) | 5 Mo (taille maximale avant avertissement) | 281 Mo |
| CodeIgniter | 67,4 Mo (binaires PHP) | 65 Mo (taille d’un nouveau projet vierge) | N/A | 132,4 Mo |

* En choisissant Django, je gagne globalement de l’espace disque par rapport à un projet Node.js, mais j’en perds considérablement par rapport à un projet PHP.
* Cependant, la taille minime du projet en lui-même permet des transferts entre développeurs plus légers dans le cadre d’un projet géré avec un contrôle de code source.

## Exécution

D’après <https://programmation.developpez.com/actu/253829/Programmation-une-etude-revele-les-langages-les-plus-voraces-en-energie-Perl-Python-et-Ruby-en-tete-C-Rust-et-Cplusplus-les-langages-les-plus-verts/>

Une image contenant table

Description générée automatiquement

* Considérant les valeurs ci-dessus, en choisissant Python, je choisis
  + Le langage le plus énergivore de mon portefeuille de connaissances. PHP consomme 2,6 fois moins, TypeScript 3,5 fois moins, JavaScript 17 fois moins.
  + Le langage prenant le plus de temps à exécuter une tâche. TypeScript est 1,5 fois plus rapide, PHP est 2,6 fois plus rapide, et JavaScript est 11 fois plus rapide.
  + MAIS c’est celui qui occupe le moindre espace dans la mémoire vive. Il en consomme à peine plus que PHP, mais presque deux fois moins que TypeScript et JavaScript.

# Fin de vie

## Sauvegarde ou suppression des données

Deux aspects à traiter :

* Un aspect technique : veux-t-on que les données soient réutilisables ou simplement détruites ?
* Un aspect juridique/éthique : la base contient des données personnelles : elles ne doivent pas être conservées au-delà de leur période d’usage. Avant la mise en fin de vie, il est donc primordial de supprimer les données personnelles des clients :
  + Se connecter au panneau d’administration (lien en bas du site)
  + Cliquer sur « Profils de clients »
  + Tout sélectionner
  + Choisir l’action rapide « Effacer les données personnelles des profils sélectionnés »  
    Une image contenant texte

    Description générée automatiquement  
    Notez que cela aura également pour effet de rendre anonyme les comptes utilisateur associés aux profils.  
    Une image contenant texte

    Description générée automatiquement

## Nettoyage

La machine hébergeant le site doit être nettoyée selon cette procédure :

1. Interrompre le processus Python qui fait tourner le site
2. Supprimer (ou déplacer ailleurs) le dossier contenant le code du projet
3. Désinstaller Python