# Page HOME : / A propos

* Mon profil (qui je suis),
* CV
* Parcours,
* Compétences,
* Liens divers

# Page Contact :

Formulaire de contact

# Page Portfolio

Liste de tous mes projets (terminés ou en cours)

# Page Rapport de Stage :

## Intro :

* Introduction sur l’entreprise (lieu, histoire, activités…)
* Contexte de travail,
* Objectifs du stage …

## Articles sur le stage :

* Intégration
* Différentes missions
* Difficultés rencontrées
* Nouvelles technos apprises…
* Compétentes utilisés
* Propositions de solutions a des problèmes
* Apprentissage
* Documentation de qualité

Liste des articles :

* + Création d’un NFT (decouverte des métadatas + du smart contact)
  + Création d’une collection de NFTs (Algorithme de création des métadatas + Smart contract dynamique)
  + Création du site de mint (React +bibliotheque etherjs)
  + Création Du générateur d’images pour NFT
  + Réalisation du site pour le voisin (a venir)

## Bilan du stage :

* Vos impressions, votre ressenti ;
* L’expérience acquise ;
* Les points d’amélioration ;

# Liste des Articles

## Introduction :

J’ai été contacté par Samuel Dubreuil, un 3D designer pour l’assister lors de la création de Nomad Gallery. Nomad Gallery est une entreprise qui a pour objectif de digitaliser le travail physique d’artistes en collaborant avec eux.

La première mission de Nomad Gallery était de digitaliser la pièce principale de l’exposition de Jean Ewen : « On the road of a nomad ».

Biographie de Jean Ewen + exposition On the road of a Nomad

L’objectif de la mission était de reproduire en 3D la voiture gravée de Jean Ewen puis de réaliser un NFT de la voiture. J’ai été donc missionné de réaliser la collection de NFTs à partir du fichier 3D réalisé par l’équipe de Nomad Gallery. L’autre mission était de m’occuper du référencement du site qui a été réalisé avant mon arrivée sur squarespace.

## Article 1 Création d’un NFT :

Pour réaliser la mission qui m’a été donné je commence par rechercher comment faire un NFT. Je suis tombé sur plusieurs solutions :

|  |  |  |
| --- | --- | --- |
| Solution 1 : | Utiliser une solution toute prête (exemple : création via OpenSea) | Créer notre propre NFT Création des metadatas du nft et du smart contract |
| Inconvénient : | Le NFT n’est pas réellement crée avant son achat et on ne peut pas l’exporter sur d’autres plateformes, il ne nous appartient pas réellement. « Lazy mint » | Les frais de mint sont chers sur la blockchain ether en ce moment (une autre solution serait d’utiliser la blockchain polygone ou d’attendre la sortie de Ethereum 2.0) |
| Avantages : | Les frais de mint (création) sont au frais de l’acheteur donc si le NFT ne se vend pas, pas besoin de payer | Le NFT nous appartient totalement. |

J’ai finalement choisi la deuxième option qui nous permet d’être plus libre avec la création des NFTs et d’automatiser leurs création, ce que nous verrons dans un second temps.

Il faut ensuite réfléchir aux metadatas (en français métadonnées). Selon Wikipédia, Une métadonnée est une [donnée](https://fr.wikipedia.org/wiki/Donn%C3%A9e_(informatique)) servant à définir ou décrire une autre donnée. Les métadonnées synthétisent des informations élémentaires sur les données

Si l’on prend l’exemple d’un document Microsoft Word, les métadonnées sont la taille du fichier, sa date de création, le nom de l’auteur, la date des différents changements ou le nombre total de modifications. On peut aussi compter le titre, les étiquettes et les commentaires.

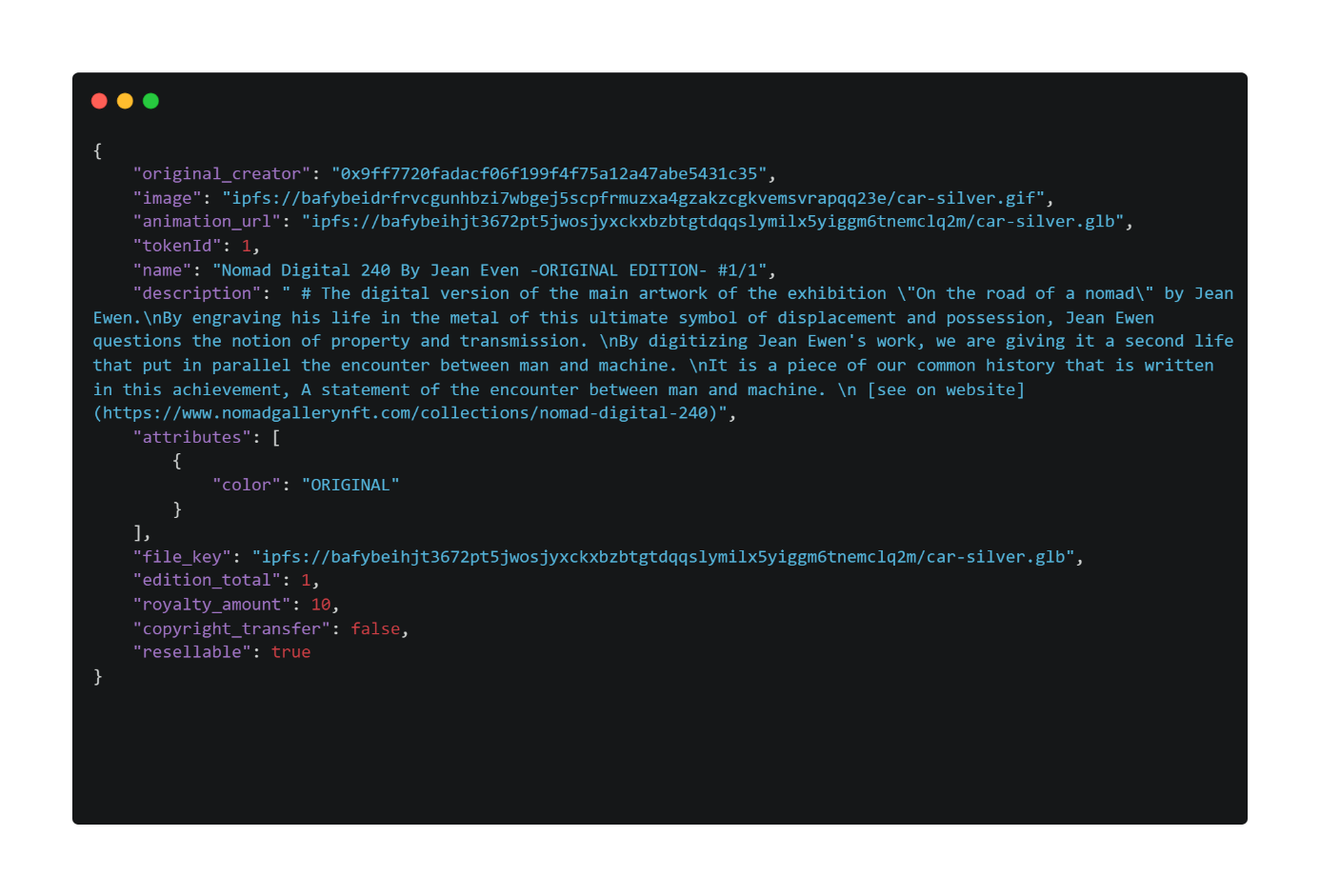
Dans le cadre des NFTs, les métadonnées jouent un rôle très important. En effet les fichier métadonnées résolvent un problème financier et technique : Il est impossible d’héberger des fichiers volumineux sur une blockchain. Par exemple un film en 1080p pèse entre 2 et 4 Go, le média Gemini estime qu’héberger 1 Go de data sur la blockchain Ethereum couterait environ 17,500 ETH. A l’heure où j’écris ces lignes, la valeur de ETH est de 1800 $ ça reviendrait à un cout de 31 000 000$ pour 1 Go.

Pour contrer ce problème, le token NFT qui est inscrit dans la Blockchain va être relié directement au fichier metadata réalisé en Json. Ce fichier va pointer vers l’url de l’artwork que l’on souhaite en NFT mais contient également les autres informations du NFT : Par exemple, si on souhaite faire un NFT d’une image, le fichier metadata va contenir l’url de l’image stockée mais également tous les autres paramètres du NFT (son nom, sa description, son créateur…).

Le standard de réalisation des fichier metadatas est le format JSON (JavaScript Object Notation).

Le fichier metadatas est stocké à l’intérieur du smart contract sous le paramètre URI (Universal Resource Identifier). On ne peut pas héberger directement le fichier Json dans la blockchain, les couts de réalisation du NFT serait trop élevé. Le paramètre URI pointe alors directement vers l’url du fichier metadata.

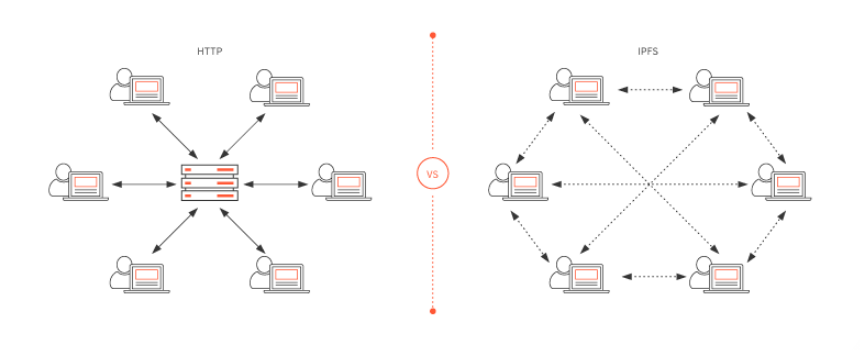
L’achat d’un NFT n’est donc pas l’achat d’un artwork mais plutôt l’achat d’un certificat d’authenticité de cet artwork.



Voici le fichier que j’ai utilisé pour réaliser les metadatas. On peut y retrouver les paramètres de notre NFT :

|  |  |
| --- | --- |
| **Paramètres :** | **Description :** |
| original\_creator | Adresse metamask du créateur du NFT, ici j’ai mis l’adresse de compte de Nomad Gallery. |
| Image | Lien de l’image uploadé sur IPFS, j’expliquerai le fonctionnement d’ipfs juste après. |
| animation\_url | Lien du gif représentant le NFT sur IPFS |
| tokenId  name | Numéro de série du NFT  Nom du NFT |
| description | Description du NFT |
| attributes | Tous les attributs particuliers du NFT. Ici on a juste besoin d’ajouter un paramètre de couleur qui va permettre de différencier la voiture Originale de la version Or. |
| file\_key | Lien du fichier téléchargeable de la voiture en haute qualité sur IPFS. |
| edition\_total | Nombre total de NFTs dans la collection. |
| royalty\_amount | Taux de droits d’auteurs qui nous sont reversés a chaque vente de NFTs. |
| copyright\_transfer | Nous ne voulons pas vendre les droits d’exploitations du NFTs donc on laisse ce paramètre sur false |
| resellable | Le NFT peut effectivement être revendu. |

Les metadatas ainsi que les medias des nft sont hébergés hors de la blochain, sur un service similaire à Google Drive. Pour les NFTs on a tendance à utiliser IPFS

IPFS (InterPlanetary File System) est un système distribué de fichiers pair à pair qui ne dépend pas de serveurs centralisés. L’objectif de ce système est de décentraliser la distribution de fichiers. 

Ce système a également pour objectif de résoudre les problèmes de http (Tous les fichiers sont centralisés dans les serveurs et sont vulnérables aux crashs et aux attaques DDOS. Les conséquences de ces vulnérabilités sont les vols de données, les fuites ou bien la perte de ces données).

Pour ajouter des fichiers sur ipfs, il faut télécharger l’application IPFS puis ajouter les fichiers que l’on souhaite dessus. Ensuite il faut récupérer l’adresses des fichiers uploadés et l’ajouter dans les metadatas

Une image contenant texte

Description générée automatiquement

Vous pouvez retrouver plus d’informations à propos d’ipfs [ici](https://cryptoast.fr/interplanetary-file-system-ipfs-reseau-partage-fichiers-web-3/)

Pour pouvoir créer le NFT à partir du smart contract, il faut installer metamask.

Metamask est un wallet (portefeuille) de [cryptodevises](https://www.comparateurbanque.com/epargne/trading/) basé sur la machine virtuelle Ethereum (Ethereum Virtual Machine). Il est donc compatible avec n’importe quelle Blockchain basé sur ether. Pour installer metamask, il suffit de télécharger l’extension metamask sur son navigateur puis de créer un compte.

Déploiement du smart contact : On utilise L’IDE Remix pour cela :

Remix est un IDE en ligne qui permet de simplement lancer et déployer des applications sur la blockchain. Ici pour tester nous n’allons pas lancer le smart contract sur la blockchain classique mais plutôt sur une blockchain de test.

Un testnet est un réseau alternatif d'une blockchain servant d'environnement bac à sable pour que les développeurs puissent mettre à l'épreuve leur code avant le déploiement sur le mainnet.

Nous allons utiliser la testnet Rinkeby

Pour utiliser la blockchain de test, il faut récupérer des Ethers de test.

<https://www.rinkeby.io/#faucet>

Il suffit d’entrer l’addresse d’un compte metamask sur le site de rinkeby pour avoir accès a des Ethers sur cette Blockchain. Ces tokens n’ont aucune valeur et permettent juste de tester des smarts contracts avant leur déploiement.

Revenons au NFT.

Pour notre NFT, nous allons utiliser le token ERC-721. Le token ERC-721 a pour caracteristique le fait d’être unique, c’est ce qui fait la valeur du NFT, de plus ce token est impossible à contrefaire. Ce token a également la possibilité d’avoir une valeur différente d’un autre token du même smart contract. Cela permet par de créer une collection de NFTs avec plusieurs variantes d’artworks pour créer de la rareté dans les NFTs par exemple. Cette caractéristique n’existe pas dans le cadre du token ERC-20, c’est pour cela qu’il faut bien choisir le token pour ce type de Projet.

Le token est paramétré lors de la création du smart contract.

Il faut donc créer le Smart Contract puis le déployer sur la blockchain.

Explication du fonctionnement du smart contract

Une fois le contrat créé, il faut le compiler sur l’IDE Remix, Remix est un IDE en ligne qui permet de compiler des smarts contracts sur la blochain Ethereum.

Ici on va le compiler sur la blockchain de test

Une image contenant texte

Description générée automatiquement

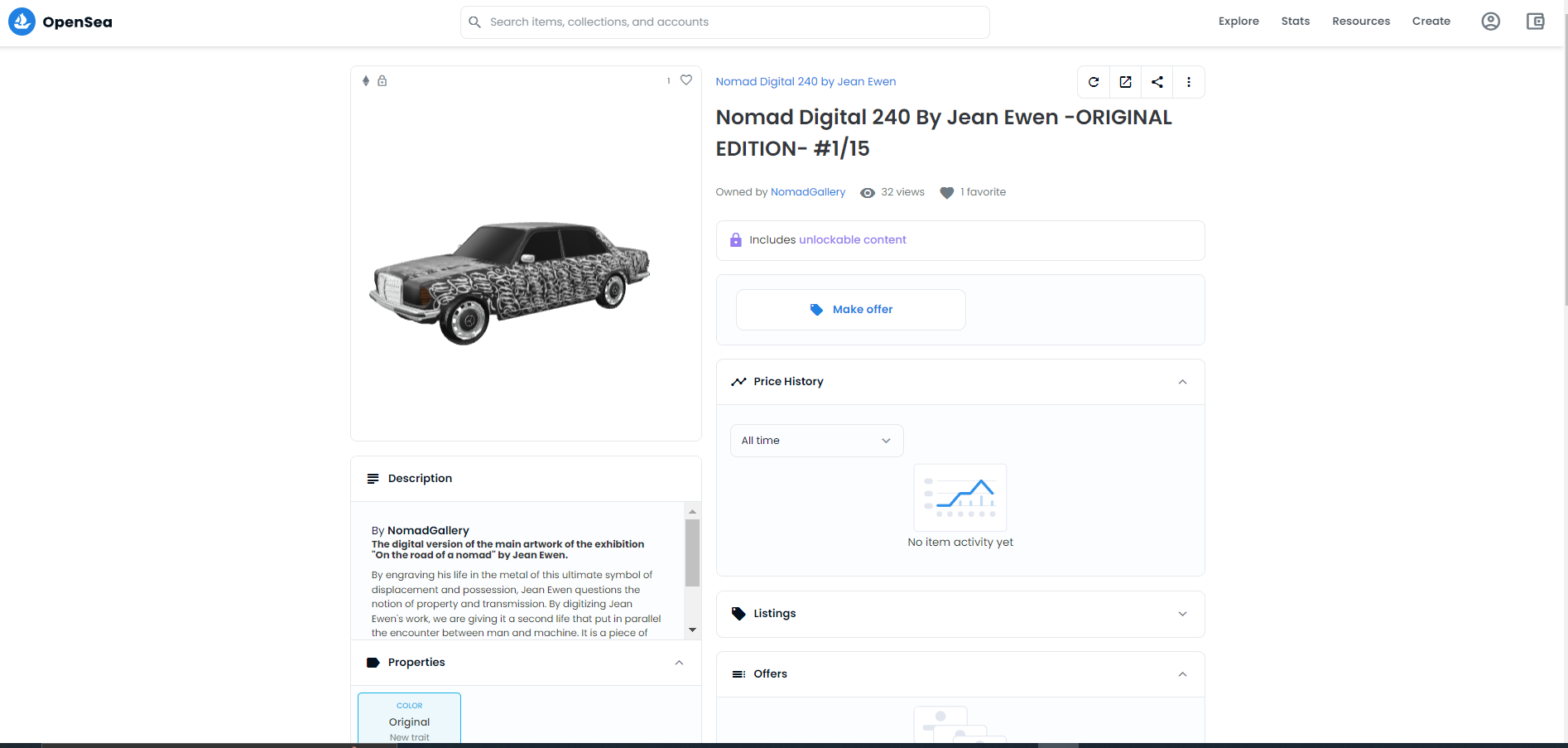
Ensuite il faut déployer les NFTs sur la blochain en entrant les paramètres du NFT( Le nom de la collection, le lien des fichiers metadatas uploadés sur IPFS et le symbole du NFT).

Une image contenant texte

Description générée automatiquement

Il faut valider la transaction sur metamask et voilà, le NFT est déployé.

On peut ensuite visualiser la transaction sur Etherscan et voir le NFT sur le profil OpenSea lié au metamask



## Article 2 : Création d’une collection de NFT :

Le problème avec la manière de créer des NFT de l’article 1 c’est que si je veux créer un grand nombre de NFTs, c’est trop long et fastidieux. J’ai donc décidé d’automatiser la création de ces NFT

A partir du précédent projet il faut trouver un moyen d’automatiser la création de plusieurs NFT.

J’ai choisi d’utiliser un script Python qui va générer les metadatas des fichiers NFT à partir d’une image et des paramètres choisit par l’utilisateur.

Le script commence à générer les metadatas sous forme de fichier JSON à partir des paramètres choisis par l’utilisateur.

Explication du script pour générer les fichiers json

Une fois les fichiers crées j’ajoute au script une partie pour pouvoir upload les fichiers automatiquement sur ipfs.

Explication de la partie upload ipfs

Une fois les fichiers json uploadés il faut créer le smart contract

Explication du smart contract

Une fois le smart contract réalisé, je déploie tous les NFT avec Remix comme pour l’exemple précedent.

Fonctionnement de REMIX et de la blockchain testnet

On peut ensuite visualiser la transaction sur Etherscan et voir le nft sur le profil OpenSea lié au metamask. Avec cette méthode nous avons pu réaliser l’entièreté de la collection de Jean Ewen sur Opensea

## Article 3 : Création du site de mint (React +bibliotheque etherjs)

La collection a bien été réalisée et les NFTs de la voiture ont bien été créés. C’est à ce moment que nous affrontons notre plus gros problème : nous n’arrivons pas à créer de l’engouement autour de ce projet. Dans le monde des NFTs, il y a 2 façons de créer de la hype et de la valeur autour d’un projet : Le projet contient une base technique qui permet à celui qui achète le NFT de le rentabiliser. On peut prendre l’exemple des jeux Play To Earn qui permettent de gagner de l’argent en jouant a un jeu, le ticket d’entrée du jeu étant le NFT.

L’autre manière de créer de l’engouement est de vendre un projet artistique. Le créateur Beeple arrive à vendre des NFTs à plusieurs millions d’euros en étant un artiste connu.



Le problème du projet Nomad Mercedes et que son créateur Jean Ewen n’est pas suffisamment connu et que sa communauté n’est pas réceptive à la mode des NFTs.

Nous avons donc essayé une nouvelle approche : Attirer les gens de la communauté Crypto. En recherchant les projets qui marchent sur Twitter, j’ai vu plusieurs personnes qui mettent en places des platformes de free mint, c’est-à-dire des platformes ou les NFTs sont gratuits mais les frais de mint (création) sont a la charge de l’acheteur.

Après discussion avec le reste de l’équipe, j’ai pensé que si nous arrivions à créer de l’engouement autour de ces NFTs gratuits, nous pourrions créer une communauté qui seraient intéressés à l’idée de d’acheter nos NFTs.

Pour réaliser ce projet, il faut réaliser un site qui va permettre de mint gratuitement les NFTs.

L’avantage avec ce site est que les acheteurs iront eux même faire le mint du NFT, ils nous achètent le NFT au prix convenu et en plus ils payent les frais de mint. Comme ça si aucun NFT n’est vendu, nous n’aurons payés aucun frais de gaz.

Description du nouveau smart contract

Ensuite nous allons commencer à créer le site de mint, pour cela j’ai décidé d’utiliser React couplé avec la bibliothèque etherjs qui permet d’interagir avec un smart contract.

J’aurais pu utiliser un autre Framework pour le site de mint mais j’ai choisi react car c’est le framework le plus abouti en termes d’utilisation pour le web 3 et c’est une technologie que je souhaite découvrir.

Explication du code réact

Pour relier le smart contact avec le site, on utilise Etherjs :

En réalité avec react il existe deux bibliothèques pour le web 3, Etherjs et web3.js mais la bibliotheque ether.js est plus complète, c’est pour cela que je vais l’utiliser.

Explication de la partie EtherJs

Ensuite j’ai dû trouver un moyen d’héberger le site, si possible gratuitement. J’ai utilisé FireBase pour cela :

Fonctionnement de FireBase

Voilà le site de mint est maintenant prêt à être utilisé. Après le lancement du site de mint nous avons essayé de le promouvoir sur les différents réseaux sociaux mais rien à faire. Impossible de créer de l’engouement autour du projet. Avec du recul ce fut la partie la plus frustrante du stage, tout le travail était fait mais impossible d’avoir un retour sur investissement. Sans compter le fait que vers la fin du projet, la mode des NFTs a commencé à s’essouffler. Peut-être que si nous étions arrivés plus tôt sur le marché, le projet aurait pu prendre.

## Article 4 Création d’un genérateur d’image NFT :

Après l’échec du site de free mint, nous nous sommes concertés. Le projet a fait perdre beaucoup d’argent a l’entreprise et il n’y a pas eu de retour sur investissement : aucun NFT n’a été vendu. Nous sommes arrivés à la conclusion qu’il fallait changer de modèle économique. Nous avons donc pensé à revoir notre formule. Si un artiste veut collaborer avec nous, alors nous lui fournirons une collection de NFT entière et déployé sur la blockchain mais nous ne nous occuperons pas de la partie vente des NFTs. Cette façon de faire nous permettra alors d’être payé a chaque réalisation peu importe la réussite du projet ou non. De plus, cela va nous permettre de proposer d’autres services plus simples, comme une réalisation de sites web ou bien de la modélisation 3D. Ce genre de services peut être mis en place sur le site web de Nomad Gallery mais également sur Fiverr. Afin de proposer nos services sur ce site, nous avons commencé par créer un compte fiverr puis nous avons réfléchi à toutes les offres que nous pourrions proposer.

Liste des offres Fiverr

Afin de pouvoir créer le service de collection d’image NFT différentes, j’ai dû réaliser un générateur d’images à partir d’assets et la suite adapter le script précédent pour l’upload de NFTs pour l’intégrer dans celui-ci.

Pour cela il faut générer toutes les images puis les metadatas à partir de ces images.

La première étape du générateur d’images est de calculer le nombre d’images maximum pouvant être réalisés avec les assets fournis.

Explication du calculateur d’images

Le script va ensuite demander à l’utilisateur le nombres d’images qu’il veut générer puis va lancer la génération.

La création d’images commence par aléatoirement relier les attributs entre eux, les images ne sont pas encore créées mais on ajoute dans un tableau la liste de tous les attributs d’une image. Ce tableau servira par la suite à créer les metadatas liés aux images.

Explication du code

A partir du tableau et des différents attributs, les images vont être crées.

Explication du générateur d’images

Une fois les images créées, le programme demande à l’utilisateur s’il souhaite ajouter metadatas aux images précédemment crées. Les images vont alors être uploadés sur ipfs et les metadatas vont être générés.

Contrairement aux précédents fichiers de métadatas, celui-ci devra contenir les attributs de l’image (dans l’exemple d’un personnage, on peut imaginer ajouter un attribut pour chaque vêtement.)

Exemple d’attribut.

Ensuite la génération des fichiers est relativement similaire à celle réaliser dans les autres articles

Explication de la création de metadatas.

Le script va upload les métadatas sur ipfs, le script est similaire à celui réalisé précédemment dans l’article sur la création de la collection de NFTs. J’intègre donc le script précédent dans le nouveau script.

Explication du script ipfs

Pour finir je dois recréer un site de mint pour la collection de NFTs. Le processus est identique a celui de l’article sur la création du site de mint. J’adapte donc le smart contract de la DAPP. Il n’y a pas beaucoup de paramètres à modifier.

Explication du smart Contract

Ensuite pour la DAPP j’ai repris le code précédemment réalisé, j’y ait integré le nouveau smart contract puis j’ai revu le design du site pour qu’il colle avec le design de collection NFT.

Explication du design

Pour le design, j’ai imaginé une webpage utilisant quelques images qui ont été générés afin d’avoir un rendu coloré.

Ensuite il ne reste plus qu’à upload le site sur FireBase comme réalisé précédemment.

Une fois le générateur réalisé, il faut créer le compte fiverr et mettre les annonces.

Screen fiverr

J’en ai également profité pour mettre les artworks réalisés sur le site web.

Screen web

## Conclusion :

Conclusion

Pour conclure ce stage