Dossier Projet

Pour le titre professionnel développeur Web / Web Mobile

S.O.M.M.A.I.R.E

[Introduction 3](#_Toc75809049)

[Résumé du dossier 3](#_Toc75809050)

[Première partie 3](#_Toc75809051)

[Seconde partie 3](#_Toc75809052)

[Premier projet 4](#_Toc75809053)

[Présentation de l’entreprise 4](#_Toc75809054)

[Présentation du projet 4](#_Toc75809055)

[Les cibles 4](#_Toc75809056)

[Enjeux 4](#_Toc75809057)

[Inventaire technique 5](#_Toc75809058)

[Principe du projet 5](#_Toc75809059)

[Cas d’utilisation 6](#_Toc75809060)

[Algorithme du script 7](#_Toc75809061)

[Exemple de document en anglais 8](#_Toc75809062)

[Second projet 10](#_Toc75809063)

[Présentation du jeu 10](#_Toc75809064)

[Présentation du projet 10](#_Toc75809065)

[Les cibles 10](#_Toc75809066)

[Spécifications fonctionnelles 10](#_Toc75809067)

[Critères 10](#_Toc75809068)

[Inventaire technique 11](#_Toc75809069)

[Arborescence du site 11](#_Toc75809070)

[Diagramme UML 12](#_Toc75809071)

[Maquettes de la partie Desktop 12](#_Toc75809072)

[Maquettes de la partie Mobile 15](#_Toc75809073)

[Extrait de code 18](#_Toc75809074)

# 

# Introduction

## Résumé du dossier

### Première partie

Lors de mon stage pour la société Ibloo (installateur de fibre optique depuis 2019, basé à Tours) j’ai eu pour projet de créer une fonction qui récupère dans un rayon de 100 mètres, les informations des bâtiments autour une coordonnée se situant dans un json.

Cette fonction a pour but d’être utilisé pour vérifier l’éligibilité à la fibre optique de potentiel client.

### Seconde partie

Smite est un jeu vidéo de type MOBA (Multiplayer Online Battle Arena) développé et édité par Hi-Rez

Studios. Le développement de ce jeu a commencé en 2011 et il est sorti officiellement sur

PC, en 2014.

Le projet a pour objectif la sélection aléatoire d’un personnage de l’univers de Smite, ainsi que les objets que le joueur va devoir acheter au cours de la partie.

Cette application personnelle pourra possiblement s’étendre à un plus large public.

Premièrement, j’ai dû établir les spécifications fonctionnelles de l’application.

En second lieu, j’ai défini les spécifications techniques, ce après quoi il a fallu créer la base de données de l’application, avec toutes les liaisons nécessaires à son bon fonctionnement.

De plus il a fallu créer la maquette, choisir le design et les couleurs du site.

Ainsi, j’ai effectué le développement du site en commençant par la partie front, tout en respectant le mockup que j’avais mis en place.

Ce dossier comprend donc : L’expression des besoins, les spécifications fonctionnelles, les spécifications techniques, les documents liés à la création du site (mockup, arborescence, documentation…) et enfin, un extrait d’une documentation anglophone, que j’ai traduit moi- même, afin de l’utiliser pour la création de mon application.

# Premier projet

## Présentation de l’entreprise

Ibloo est une entreprise qui installe la fibre optique partout en France pour le particulier et le professionnel depuis 2019.

L’entreprise étant basé à Tours, mon stage à donc été entièrement en distanciel.

## Présentation du projet

Ce projet est mis en place pour les potentiels futurs clients puissent vérifier s’ils sont éligibles à la fibre optique.

Ma partie est la fonction qui permet de récupérer les bâtiments dans un rayon de 100 mètres de l’adresse entré par le client. Cette fonction est réalisée en Python, j’ai donc dû apprendre le Python ainsi que d’utiliser certain modules Python.

### Les cibles

Ce projet a pour cible les potentiels futurs clients qui souhaitent obtenir la fibre optique.

### Enjeux

1. Apprentissage et utilisation du langage Python.
2. Recherche dans les documentations (Notamment en anglais).
3. Utilisation des types de projections géographiques.

## Inventaire technique

**Python 3.9.5:** Utilisé pour la réalisation de la fonction

**Visual Studio Code :** L’éditeur de texte pour le développement du site

**Trello :** Pour l’organisation du projet

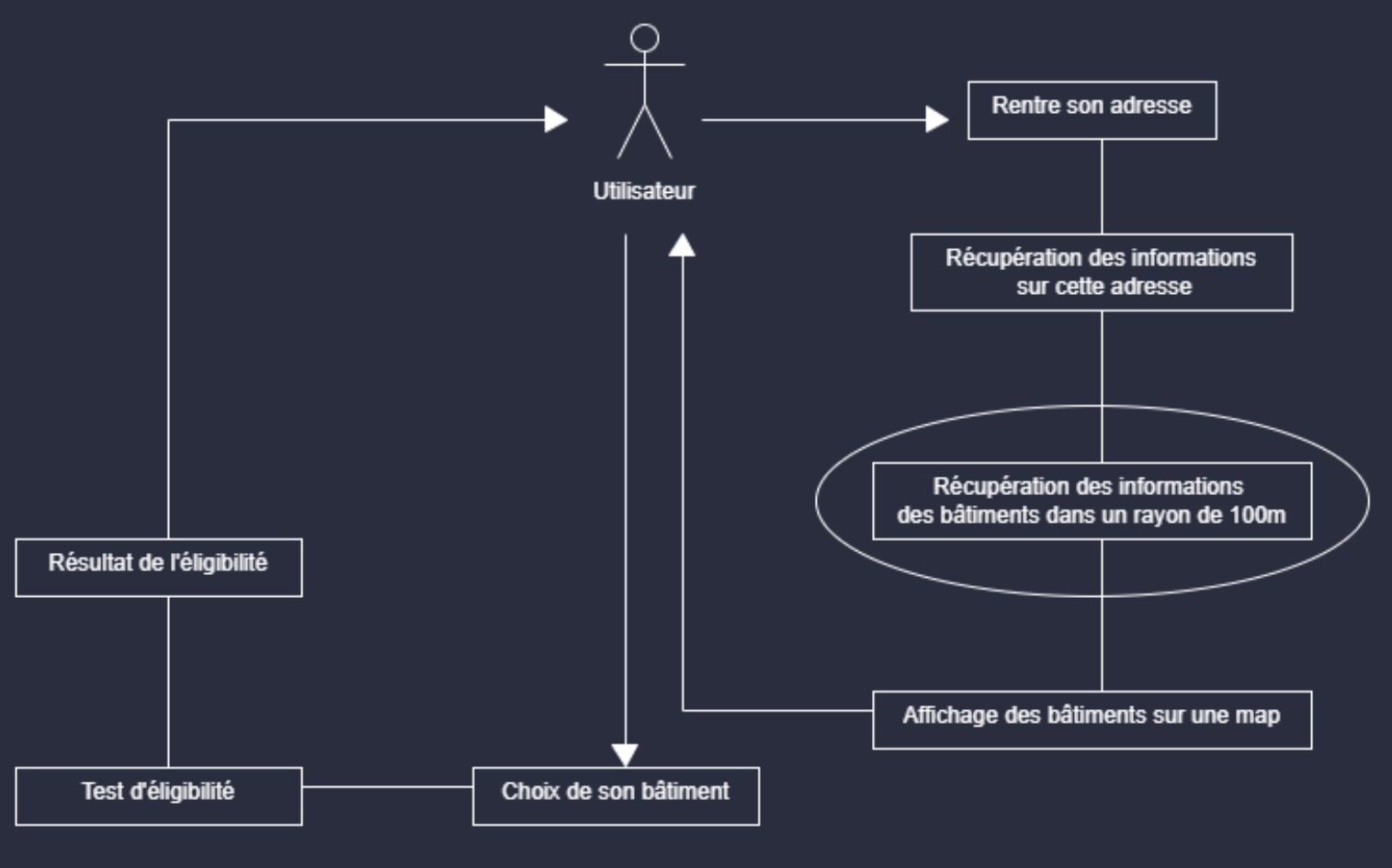
**Geo Data Viewer :** Pour visualiser sur une map.

## Principe du projet



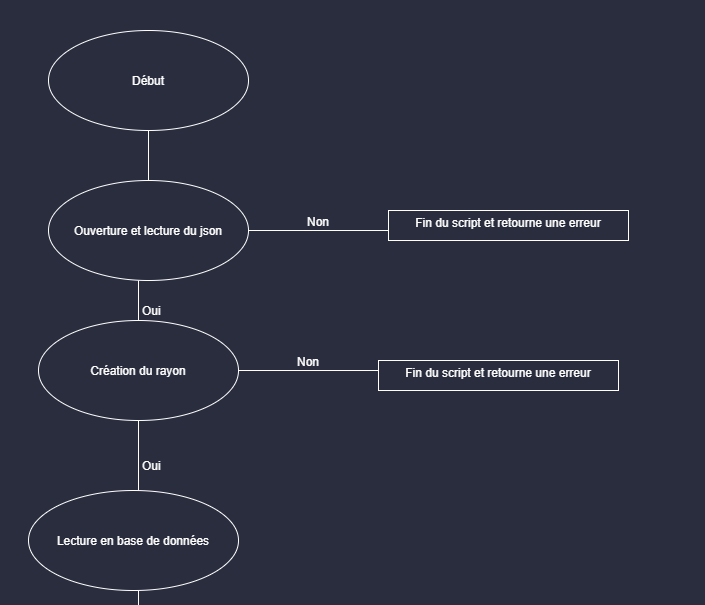
Pour ce projet je n’ai travaillé que sur une seule fonctionnalité et en Local. J’ai donc dû me servir d’un json de test au lieu d’avoir accès directement à l’API Géo-Gouv.

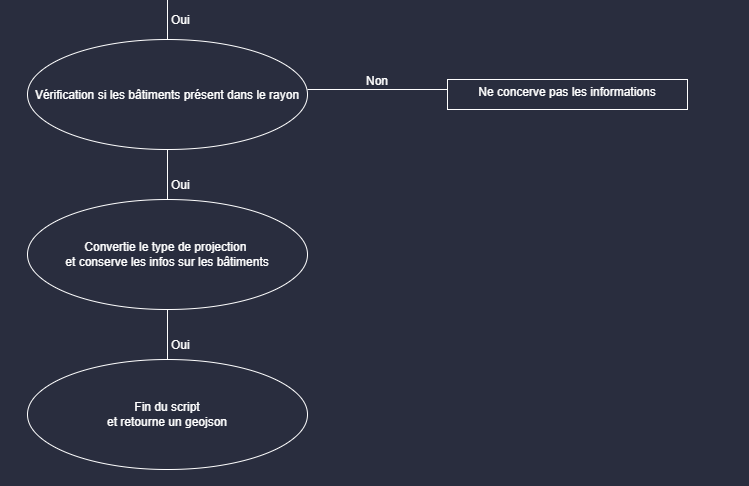
## Cas d’utilisation



L’utilisateur rentre son adresse, l’API gouvernementale Géo-Gouv va alors envoyer un json avec des coordonnées qui corresponds à celle de l’adresse rentrer par l’utilisateur. Une fonction va ensuite récupérer les informations des bâtiments dans un rayon de 100 mètres. Suite à cela l’utilisateur verra sur une map les bâtiments et devra choisir son bâtiment. Une fonction va vérifier ensuite s’il est éligible à la fibre et renverra à l’utilisateur le résultat.

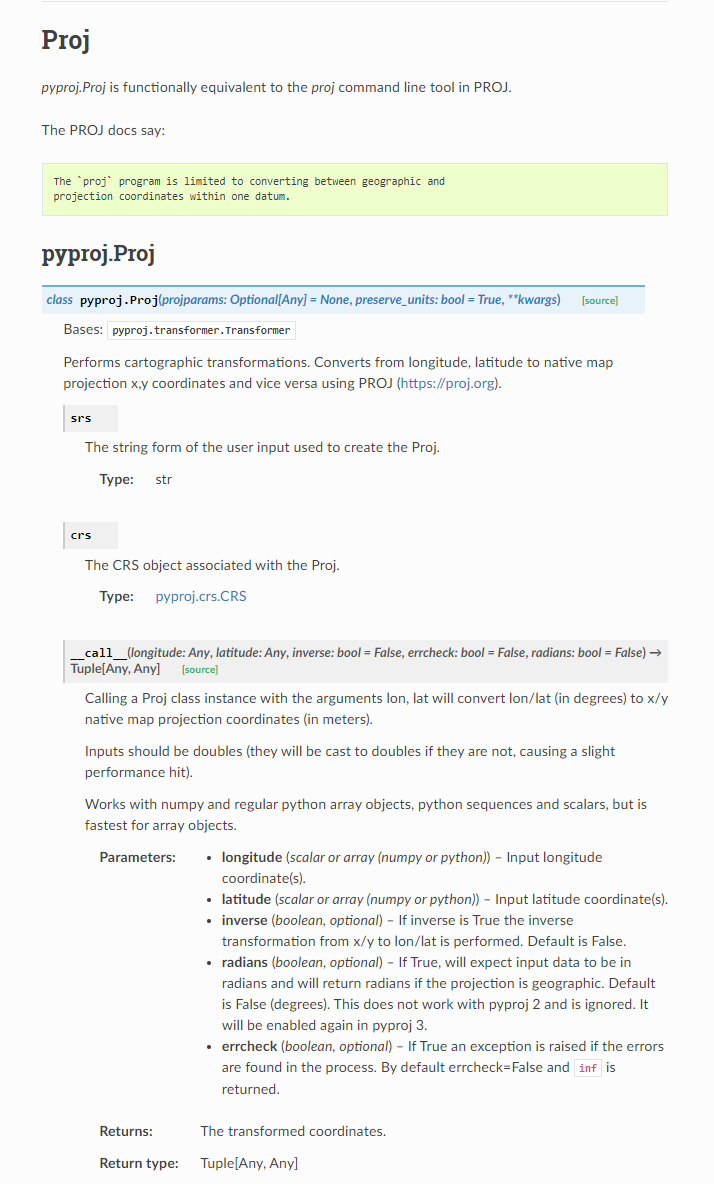
## Algorithme du script





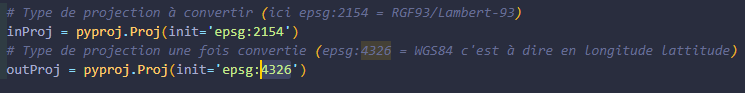
## Exemple de document en anglais

Je me suis servi de plusieurs documents afin d’apprendre le Python et pour exploiter des modules. Je vais donc donner comme exemple la documentation de la fonction Proj du module Pyproj.



Cette fonction sert à convertir les projections géographiques. J’ai dû m’en servir pour convertir du RGF93 (Projection géographique Française) en WGS 84(Projection géographique Mondial)

*Variable de préparation à la conversion*



*Conversion de projection*



# Second projet

## Présentation du jeu

Smite est un jeu vidéo d'arène de bataille en ligne multijoueur en vue à la troisième personne où les joueurs incarnent des divinités. Développé et édité par Hi-Rez Studios, le jeu est sorti le 25 mars 2014. Ce jeu servira de support pour ce projet.

## Présentation du projet

Ce projet est mis en place afin de créer des challenges sur le jeu vidéo « SMITE ». Cette application choisis aléatoirement le personnage à jouer ainsi que les objets qu’il devra acheter au cours d’une partie.

Il y a aussi une partie Back office avec la gestions complètes des divinités.

### Les cibles

Ce site s’adresse aux joueurs ayant un minimum de connaissance sur le jeu, sur les dieux mais aussi sur les objets.

### Spécifications fonctionnelles

Tirage aléatoire d’un personnage avec ces objets

Gestion complète des personnages stockées en base de données :

* + Insertion de données
  + Mise à jour de données
  + Suppression de données
  + Affichage des données
  + Recherche des données

### Critères

L’écriture du code a été réalisé en POO (Programmation Orienté Objet).

## Inventaire technique

**HTML 5 :** Utilisé pour la version statique du site web.

**CSS 3 :** Pour l’habillage du site.

**Javascript :** Pour la rédaction du dossier projet.

**PHP 7.4.9 :** Utilisé pour rendre le site dynamique et pour la création des fonctions.

**Google Chrome et Opera GX :** Pour les différents test et l’optimisation du site.

**Visual Studio Code :** L’éditeur de texte pour le développement du site.

**Git :** Pour le contrôle des versions.

**GitHub :** Pour le stockage dans le cloud.

## Arborescence du site

Index

Administration

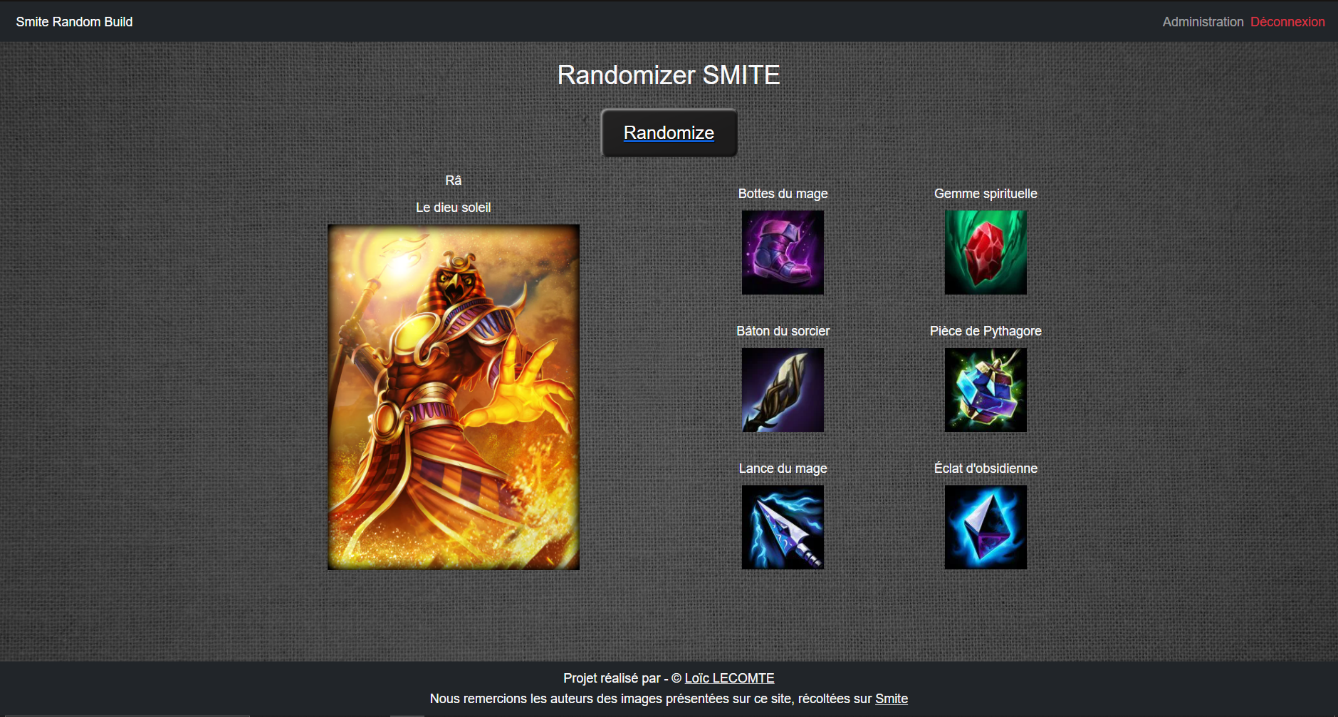
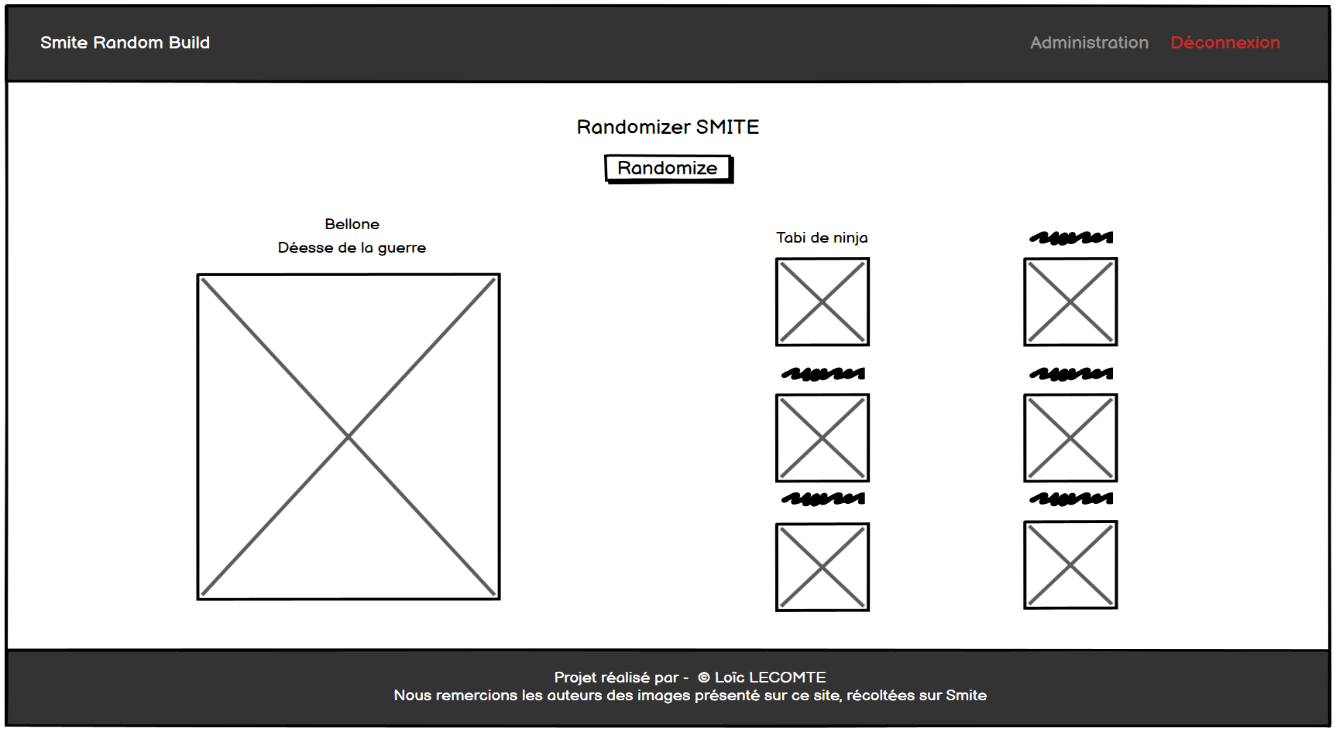
Tableau de bord

## Diagramme UML

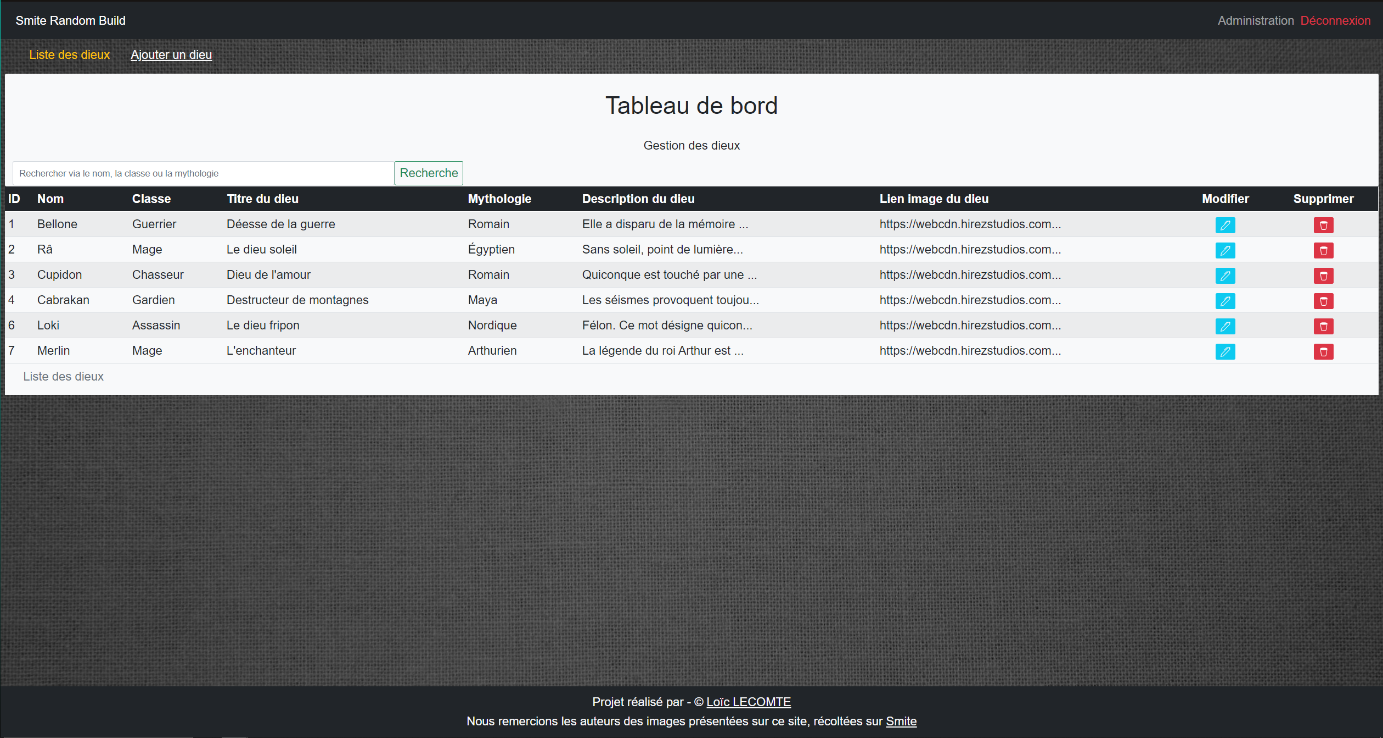
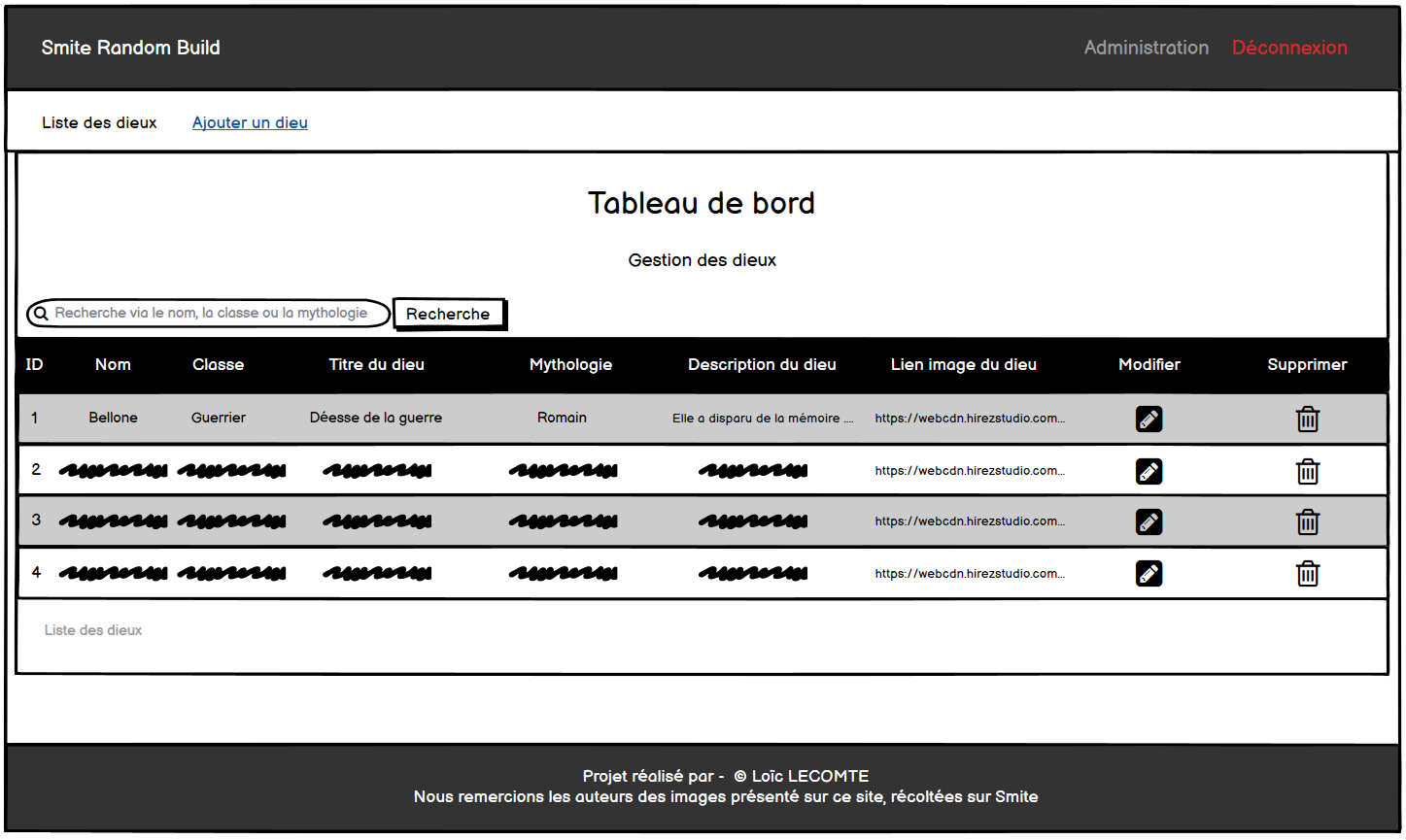


## Maquettes de la partie Desktop

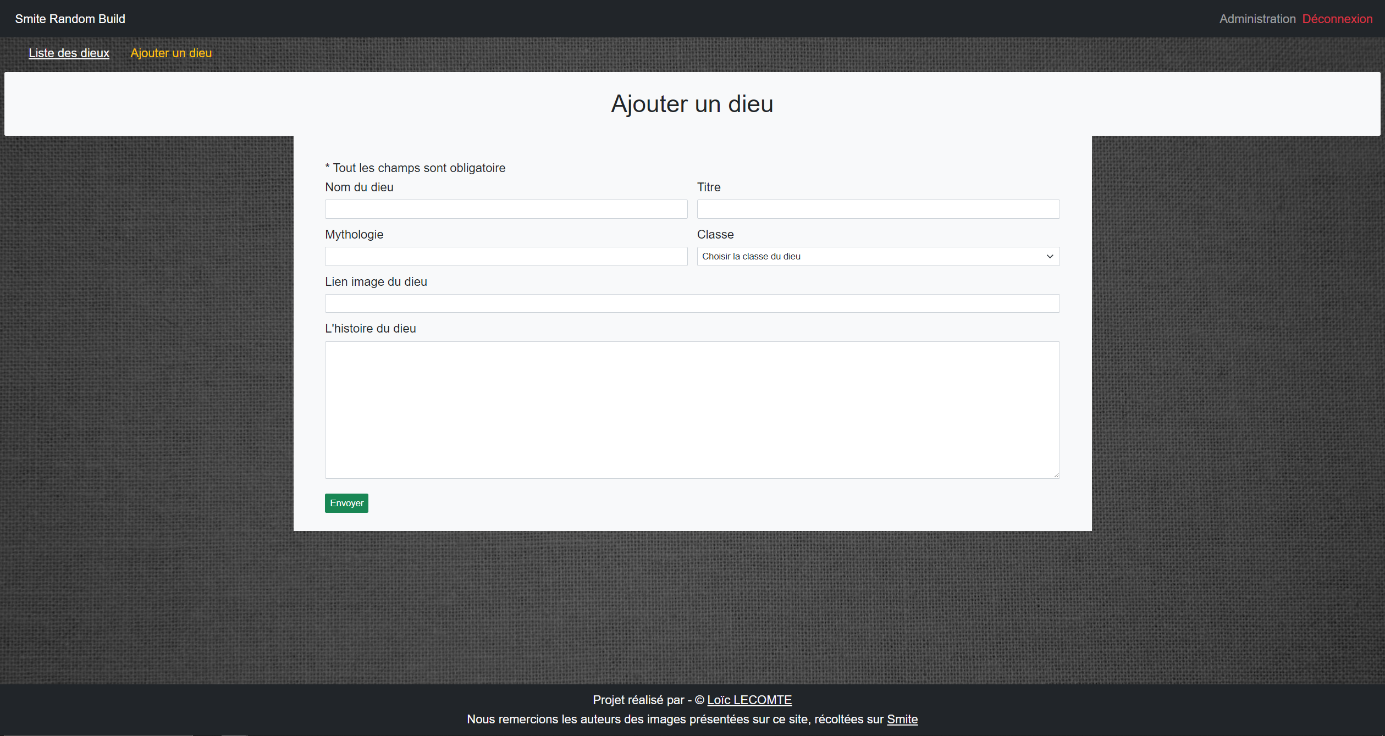
La page d’accueil



La page d’admin

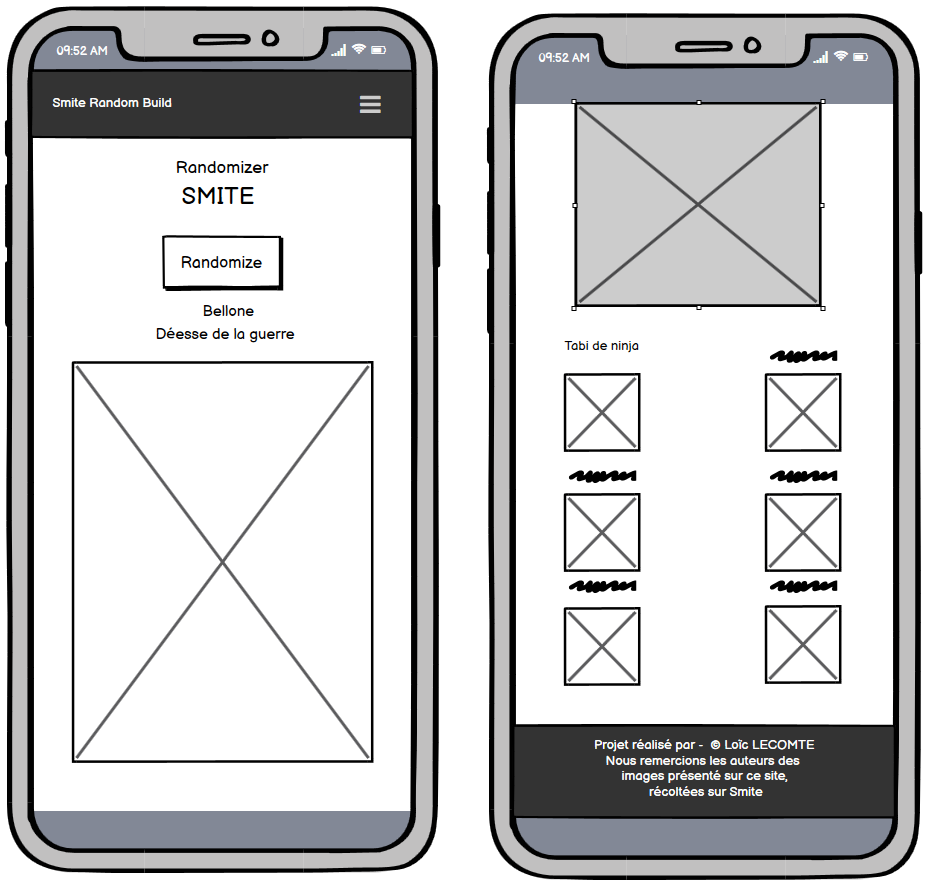


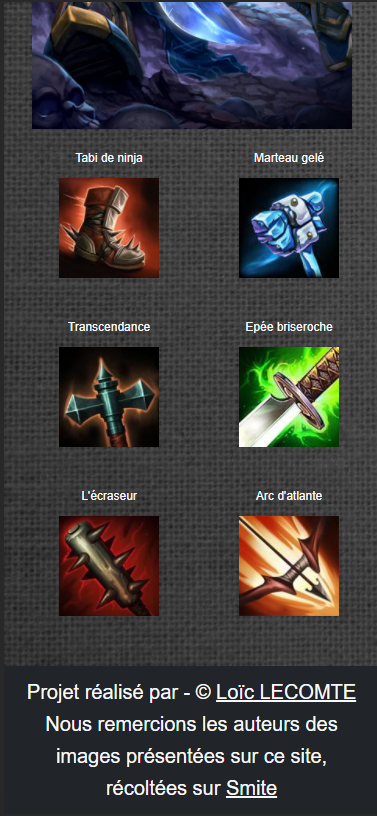
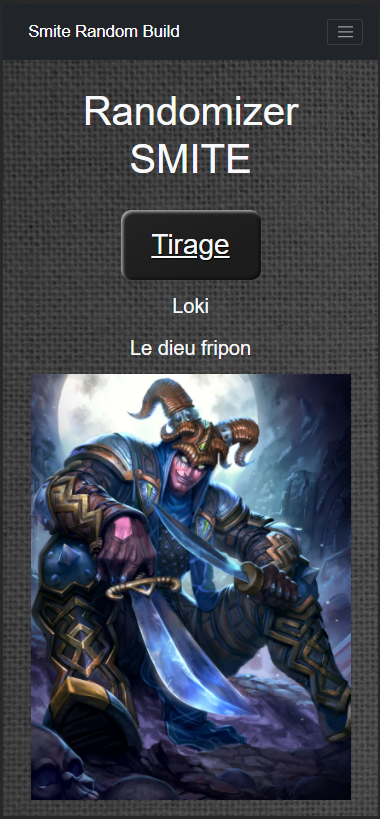
Formulaire d’ajout d’un personnage



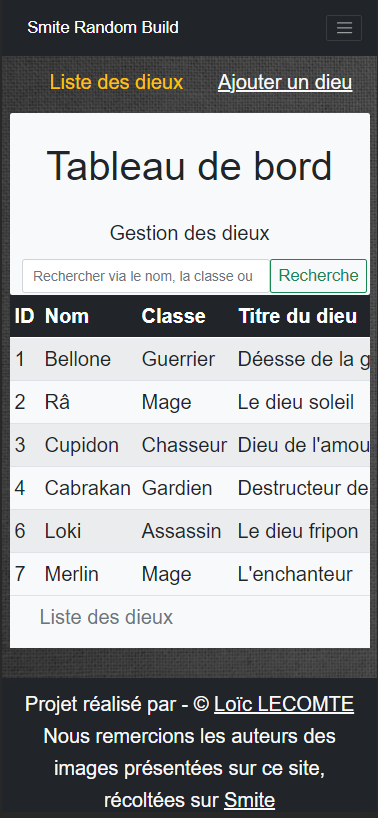
## Maquettes de la partie Mobile

La page d’accueil

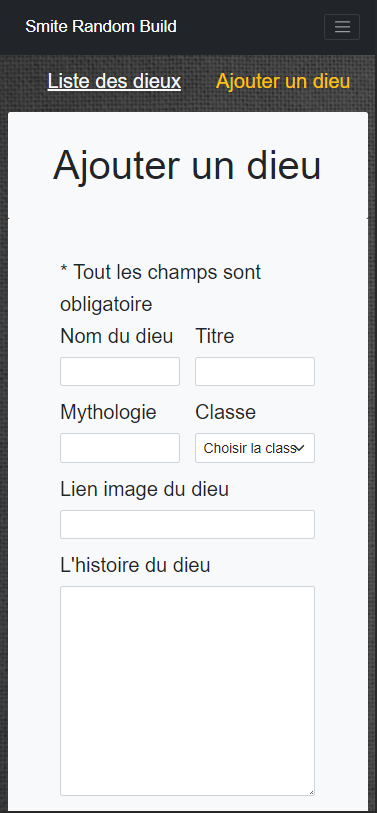
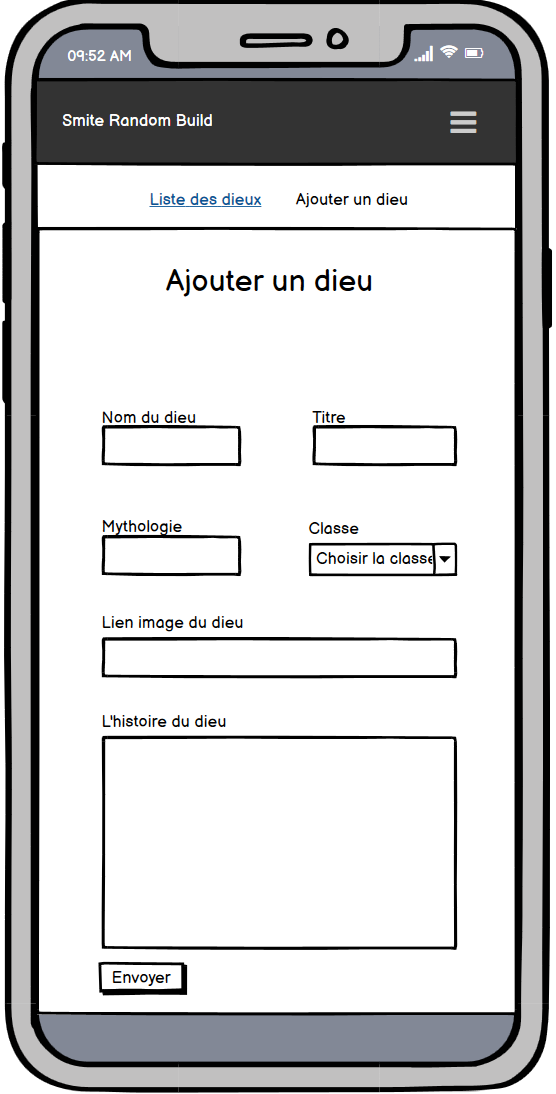




La page d’admin



Formulaire d’ajout d’un personnage



## Extrait de code

Extrait de Javascript représentant l’affichage d’une info bull lors du focus sur une image



Objectif à atteindre

Concevoir et réaliser une fonction regroupant des données récupérées via une API et une base de données pour être réutiliser pour de futures fonctions.

Critères

Utilisation du langage Python

Présentation de l’application

Objectif du projet

Création d’un lecteur de musique grâce a des API de service de musique (Spotify pour commencer) avec un traitement des données (comme nos playlists) en base de données.

L’application étant un blog, actuellement, la page d’accueil présente les derniers articles postés, avec la possibilité de cliquer pour lire l’article au complet ainsi que de le commenter.

Pour accéder à l’accueil, l’utilisateur doit se connecter sur son compte, s’il n’en possède pas, un lien est présent pour le créer.

Fonctionnalités attendues

Possibilité d’insérer, de mettre à jour, de supprimer ou de lire les données en base de données comme les comptes.

Spécifications fonctionnelles

Gestion complète des catégories stockées en base de données :

Insertion de données

Mise à jour de données

Suppression de données

Affichage des données

Recherche des données

Maquette, conception de l’interface

Page pour se connecter

Page pour créer son compte

Modèle de données



La partie gestion de compte comporte 2 tables :

* La table « **users** » pour les données utilisateurs
* La table « **roles** » pour assigner des rôles aux utilisateurs

Les deux tables sont reliées via une clé primaire id\_role (dans la table roles) à une clé étrangère id\_role (dans la table users).

Un utilisateur a toujours un rôle, et seulement un rôle (cardinalité 1, 1) or un rôle peut ne pas avoir d’utilisateur ou en avoir plusieurs.

Par défaut, un nouvel utilisateur aura le rôle « **User** ». Son rôle pourra être modifié par la suite.

Tests

Seul un administrateur connecté peut gérer les comptes utilisateurs.

|  |  |  |
| --- | --- | --- |
| Module | Tests | Résultat du test |
| Afficher les comptes | On vérifie qu’il y a bien des comptes dans la base de données si oui on les affiche dans un tableau | On affiche toutes les informations de tous les comptes présents en base de données (à l’exception du mot de passe) |
| Ajouter un compte | On a un formulaire pour créer un compte | * Si le formulaire est vide, on affiche un message d’erreur demandant à l’utilisateur de remplir le champ. * Si le pseudo ou l’adresse email existe déjà en base de données, alors un message d’erreur apparait. * Si les deux champs pour le mot de passe ne corresponds pas, alors un message d’erreur apparait. * Sinon, le compte est créé et stocké en base de données, et l’utilisateur est redirigé vers la page de connexion. |
| Modifier un compte | On clique sur le bouton pour éditer un compte, on est redirigé sur un formulaire, remplie par défaut par les infos du compte sélectionné (seul le mot de passe ne peut pas être modifié pour le moment) | * Si le formulaire est vide, on affiche un message d’erreur demandant à l’utilisateur de remplir le champ. * Si le pseudo ou l’adresse email existe déjà en base de données (sauf si c’est l’email et le pseudo du compte en cours de modification), alors un message d’erreur apparait. * Sinon les informations sont modifiées en base de données et l’administrateur est redirigé vers la page de gestion des comptes. |
| Supprimer un compte | On clique sur le bouton pour supprimer un compte |  |