

| CREATION D'UN SITE WEB



I/	Cahier des charges	2
1-	Descriptif de l'existant.....	2
2-	Besoins	2
3-	Contraintes.....	3
II/	Ressources.....	4
1-	Ressources mises à disposition	4
2-	Ressource nécessaire à la mise en place.....	4
3-	Gestion des ressources	5
III/	Analyse.....	6
1-	Descriptifs des solutions	6
2-	Comparaisons des solutions.....	7
3-	Choix d'une solution	8
4-	Cartographie du réseau.....	9
5-	Etude de l'impact sur le SI existant	10
6-	Phasage de l'intervention	10
7-	Prévision des tests de validation.....	11
8-	Déploiement	12
IV/	Mise en place.....	13
1-	Réalisation.....	13
2-	Rapport de tests.....	17
3-	Rapport de déploiement.....	18
V/	Bilan	18
1-	Conclusion.....	18
2-	Auto-évaluation.....	19

I/ Cahier des charges

1- Descriptif de l'existant

Dans le cadre du BTS SIO option SISR, il est demandé de concevoir et de maintenir des solutions techniques répondant à des besoins professionnels concrets. La création d'un site web personnel s'inscrit dans cette démarche : il doit présenter les compétences acquises durant les deux années de formation ainsi que les réalisations techniques produites en entreprise et en cours. Ce site constitue également un support essentiel pour valoriser un profil professionnel auprès de futurs employeurs.

L'environnement matériel et logiciel disponible avant la réalisation du projet est le suivant :

- Un PC fixe, utilisé comme poste principal pour le développement du site web.
- Un serveur provisoire basé sur un ancien PC portable Alienware 14, reconverti temporairement en serveur domestique.

Il exécute Proxmox VE, permettant la création et la gestion de machines virtuelles. Actuellement, il héberge :

- une VM dédiée au service Plex
- une future VM destinée à accueillir le serveur web pour héberger le site du portfolio.

Ce serveur n'a qu'un rôle transitoire : un PC qui aura le rôle de serveur domestique dédié est en cours d'assemblage et viendra remplacer cette solution provisoire.

- Un PC portable, utilisé lorsque le poste fixe n'est pas disponible. Il permet de poursuivre la rédaction, les tests ou la consultation du site en situation de mobilité.
- Une connexion internet via une Freebox Révolution, permettant d'accéder aux ressources nécessaires en ligne, aux dépôts de packages et aux outils de développement. Elle permet également, si besoin, d'exposer certains services à distance.
- Le logiciel Visual Studio Code, utilisé comme environnement de développement principal pour la gestion du code HTML, CSS, JavaScript et PHP.

L'existant ainsi décrit constitue une base solide pour mener à bien la création du site web : le matériel est opérationnel et l'environnement logiciel permet un développement structuré dans des conditions proches d'un contexte professionnel.

2- Besoins

La création du site web s'inscrit dans un objectif pédagogique et professionnel : disposer d'un support clair, structuré et accessible permettant de présenter l'ensemble des compétences acquises au cours du BTS SIO option SISR ainsi que les réalisations produites en entreprise.

Pour cela, plusieurs besoins ont été identifiés :

Besoins fonctionnels

- Présenter le parcours professionnel : informations personnelles, objectifs, compétences clés et présentation structurée de l'évolution en alternance.
- Mettre en valeur les projets réalisés : situations professionnelles, travaux en entreprise, procédures techniques, scripts, schémas réseaux, et tout document pertinent.

- Offrir une navigation simple et intuitive : l'utilisateur doit pouvoir accéder facilement à chaque section du site (Profil, Formations, Réalisations, Entreprise, Veille, Contact...).
- Permettre un contact direct : mise en place d'un formulaire de contact fonctionnel permettant aux visiteurs ou recruteurs de joindre l'étudiant facilement.
- Assurer l'accessibilité sur tous les appareils : le site doit être responsive, c'est-à-dire lisible et ergonomique sur ordinateur, tablette et smartphone.

Besoins techniques

- Héberger le site sur un serveur web dédié fonctionnant dans une machine virtuelle gérée sous Proxmox.
- Mettre en place une pile logicielle, notamment un serveur web (Apache ou Nginx), PHP et les modules nécessaires au fonctionnement du formulaire de contact.
- Faciliter la maintenance et les mises à jour : structure claire du projet, fichiers CSS organisés, scripts externalisés, séparation front-end / back-end.
- Assurer une bonne performance d'affichage : optimisation du poids des fichiers, animations maîtrisées, chargement asynchrone de certains scripts.

Besoins organisationnels

- Pouvoir développer et tester le site localement avant déploiement sur le serveur virtuel.
- Documenter l'ensemble des manipulations et configurations, afin de les réutiliser lors d'un déploiement futur sur le nouveau serveur domestique en cours d'assemblage.
- Rendre le site facilement modifiable en cas d'évolution du parcours professionnel ou d'ajout de nouvelles réalisations.

3- Contraintes

La réalisation de ce site web s'inscrit dans un calendrier relativement serré. Le projet doit en effet être finalisé entre avril et mai, tout en étant développé en parallèle de l'alternance. Cela limite considérablement le temps disponible pour concevoir l'architecture du site, travailler l'interface graphique, intégrer les fonctionnalités et effectuer les tests nécessaires. Les périodes de développement se trouvent ainsi restreintes aux soirées, aux week-ends et aux moments libres, ce qui exige une organisation rigoureuse pour tenir les délais imposés par le BTS.

Le projet doit également composer avec un niveau de compétence encore en construction. Les connaissances en HTML, CSS et JavaScript proviennent essentiellement des enseignements suivis en première année. Elles restent donc partielles et nécessitent un apprentissage complémentaire, notamment pour mettre en place un design responsive, structurer correctement le code ou intégrer des composants externes comme PHPMailer. La création d'un site complet, ergonomique et visuellement cohérent demande ainsi un temps d'appropriation supplémentaire afin d'acquérir les réflexes techniques indispensables.

La sécurité représente une autre contrainte importante, car le site est hébergé sur un serveur personnel. Cette configuration impose de prêter une attention particulière à la gestion des risques : configuration du serveur web, contrôle des accès, protection des ports ouverts, surveillance de l'activité et respect des permissions nécessaires au bon fonctionnement des fichiers sensibles. L'utilisation d'un formulaire de contact complique encore cet aspect, puisqu'elle nécessite l'envoi d'e-

mails via un serveur SMTP authentifié. Les identifiants doivent donc être protégés dans un fichier de configuration isolé, et les échanges doivent impérativement être chiffrés. Par ailleurs, les données saisies par les visiteurs doivent être filtrées afin d'éviter les injections et limiter le risque d'exploitation malveillante.

II/ Ressources

1- Ressources mises à disposition

Pour mener à bien la conception du site web, plusieurs ressources matérielles, logicielles et organisationnelles étaient déjà disponibles avant le début du projet. L'environnement principal de travail repose sur un ordinateur fixe, utilisé pour le développement, les tests et l'intégration du site.

En complément, un ancien ordinateur portable Alienware 14 a été reconverti de manière provisoire en serveur domestique. Celui-ci fonctionne sous Proxmox et héberge déjà une machine virtuelle dédiée à Plex. Une seconde machine virtuelle est prévue afin d'accueillir le serveur web du portfolio. Bien qu'il s'agisse d'une solution transitoire, ce serveur fournit un environnement fonctionnel pour l'hébergement, l'accès distant, la configuration réseau et les tests en conditions réelles. À terme, il sera remplacé par un PC fixe dédié (actuellement en cours de montage) mais il assure pour le moment l'ensemble des besoins nécessaires au déploiement du site.

Un ordinateur portable personnel est également disponible en cas d'indisponibilité du poste fixe. Il offre la possibilité de poursuivre le développement ou la maintenance du site même lors des déplacements ou des sessions de travail hors domicile, assurant ainsi une continuité de production.

Sur le plan des ressources réseau, l'accès Internet est fourni par une Freebox Révolution. Cette connexion garantit un débit suffisant pour travailler, télécharger les outils nécessaires, gérer le serveur Proxmox et procéder aux tests d'accessibilité du site. La box sert également de point d'entrée pour la configuration du NAT et pour l'ouverture des ports indispensables au bon fonctionnement du serveur web. Concernant les outils logiciels, le développement est réalisé avec Visual Studio Code, un éditeur moderne et extensible qui facilite la gestion du HTML, du CSS, du JavaScript et du PHP. Il permet de structurer efficacement le code, de profiter d'extensions d'aide à la productivité et d'effectuer des tests en local avant le déploiement.

2- Ressource nécessaire à la mise en place

La mise en œuvre du site web nécessite un ensemble de ressources techniques, logicielles et documentaires qui permettront d'en assurer le développement, le fonctionnement et la mise en ligne.

La première ressource indispensable est un serveur web dédié, installé sous Debian 13 au sein de l'infrastructure Proxmox. La VM n'aura pas besoin d'une configuration importante : 2 Go de RAM, 1 cœur et un disque de 10 Go. Cette machine virtuelle a vocation à héberger l'ensemble du site et doit donc disposer d'une pile LAMP comprenant Apache2, PHP et MariaDB. Même si la partie base de données n'est pas utilisée pour la première version du projet, l'installation de l'ensemble de la pile garantit la possibilité d'évolutions futures et permet de suivre les pratiques standard d'hébergement web.

Afin d'assurer l'accessibilité publique du portfolio, un nom de domaine sera réservé auprès d'OVH, puis configuré pour pointer vers l'adresse du serveur web. Cette étape est indispensable pour rendre le site

consultable depuis Internet de manière stable et professionnelle. Elle implique également une gestion correcte des enregistrements DNS ainsi que la mise en place d'une configuration réseau adaptée.

Pour accélérer la conception du site et garantir une base visuelle moderne, un template HTML/CSS/JavaScript est utilisé comme point de départ. Ce modèle est ensuite entièrement modifié et amélioré afin de correspondre aux besoins du portfolio, notamment en termes d'ergonomie, de structure, de contenus, d'animations et de compatibilité mobile. L'utilisation d'un template permet de réduire le temps de conception initial tout en assurant un rendu professionnel dès les premières itérations.

Face aux connaissances limitées en HTML, CSS et JavaScript, la réussite du projet repose également sur l'exploitation de ressources documentaires variées. Les guides officiels des technologies utilisées, les cours de développement web suivis en première année, ainsi que des ressources en ligne spécialisées constituent des appuis essentiels pour approfondir les notions ou résoudre des problèmes techniques. L'assistance AI comme ChatGPT contribue également à clarifier certains points, propose des pistes d'optimisation et aide à comprendre des comportements parfois complexes du code.

Enfin, une réflexion a été engagée concernant l'architecture réseau et la sécurité. L'hébergement d'un service accessible publiquement implique de préserver l'intégrité du réseau domestique. Ainsi, l'idée de mettre en place une DMZ (zone démilitarisée) a émergé afin d'isoler le serveur web du reste du réseau interne et de limiter l'exposition des autres machines en cas de compromission. Même si cette architecture dépendra des capacités matérielles et des équipements disponibles, cette approche démontre une volonté d'intégrer les enjeux de sécurité dès la conception.

3- Gestion des ressources

La gestion des ressources constitue un élément essentiel pour assurer la bonne conduite du projet et garantir que chaque étape de développement s'effectue dans des conditions optimales. Elle consiste à organiser les moyens techniques, matériels et documentaires de manière à éviter les interruptions, les pertes de temps ou les erreurs de configuration, tout en assurant une progression cohérente du travail.

La première phase consiste à planifier l'utilisation du serveur Debian 13 hébergé sur Proxmox. Une allocation précise des ressources (mémoire RAM, stockage et CPU) est définie afin de garantir la stabilité du serveur web tout en préservant le fonctionnement des autres machines virtuelles présentes sur l'infrastructure.

Concernant l'hébergement du portfolio, l'installation de la pile LAMP est organisée de manière méthodique : configuration du service Apache2, installation des modules PHP nécessaires, mise en place des permissions, puis sécurisation de l'accès au serveur. Cette structuration permet de disposer d'un environnement de développement stable et reproductible.

La gestion du nom de domaine OVH est également planifiée afin d'accéder au site plus facilement. Cette gestion inclut également la vérification de la propagation DNS et des redirections éventuelles.

Pour la partie développement, l'éditeur Visual Studio Code est utilisé comme outil central. Le template initial est intégré dans l'environnement local afin d'assurer une modification progressive, suivie et structurée. Une organisation claire des fichiers (HTML, CSS, JavaScript, images et scripts PHP) permet de conserver un projet lisible et facilement maintenable. La gestion documentaire repose sur la consultation régulière des ressources nécessaires : documentation Apache, PHP, guides HTML/CSS/JS,

tutoriels spécialisés et supports de cours. Cette démarche permet de maintenir une cohérence dans le code produit et d'éviter les solutions improvisées ou obsolètes. L'utilisation ponctuelle d'une IA contribue également à la résolution des blocages techniques rencontrés durant le développement.

Enfin, la gestion de l'aspect sécurité est intégrée dès le début du projet. La réflexion autour d'une future DMZ, la mise en place de permissions strictes sur les fichiers du serveur, ainsi que la gestion des identifiants dans un fichier séparé (config.php), participent à une approche responsable de l'hébergement web. Cette organisation permet d'anticiper les risques tout en préservant la stabilité et la disponibilité du service.

III/ Analyse

1- Descriptifs des solutions

Plusieurs solutions sont possibles afin de créer un site web :

- **Utilisation d'un CMS WordPress** : WordPress est un système de gestion de contenu (CMS) permettant de créer un site web sans devoir développer l'ensemble du code à la main. Une fois installé sur un serveur web, il offre une interface d'administration complète à partir de laquelle il est possible de gérer les pages, la mise en forme, les menus et les contenus. WordPress fonctionne autour d'un système de thèmes qui définissent l'apparence générale du site, ainsi que de plugins qui ajoutent des fonctionnalités avancées (sécurité, formulaires, optimisation, SEO, etc.).

L'installation nécessite un serveur web fonctionnel (Apache, PHP et MySQL), mais une fois en place, la prise en main est relativement rapide. WordPress présente l'avantage de proposer un large catalogue de modèles professionnels, ce qui permet d'obtenir rapidement un rendu visuel soigné. Cependant, sa grande flexibilité repose sur un écosystème de plugins souvent lourds ou complexes, ce qui peut entraîner une augmentation des risques de failles de sécurité, des besoins de maintenance régulière, et une dépendance à des composants tiers.

- **Solutions no-code (Google Sites, Wix, Webnode, Wix, etc.)** : Les solutions no-code permettent de créer un site web sans écrire une seule ligne de code. Elles reposent sur des interfaces de type « glisser-déposer » qui simplifient considérablement la conception. L'utilisateur choisit un modèle puis organise les éléments graphiques comme du texte, des images, des boutons ou des sections préformatées.

Ces plateformes intègrent automatiquement l'hébergement, la mise en ligne, la gestion du certificat HTTPS et même l'optimisation pour les appareils mobiles. Elles constituent donc une solution accessible, rapide à mettre en œuvre et adaptée à une personne ayant peu de compétences techniques. Toutefois, elles présentent des limites importantes : la personnalisation est restreinte, certaines fonctionnalités avancées sont payantes et la structure du site reste le plus souvent figée. Par ailleurs, l'utilisation de ce type de solution limite fortement l'apprentissage technique et ne permet pas de valoriser des compétences en administration, en développement ou en intégration web, pourtant attendues dans le cadre du BTS SIO SISR.

- **Développement manuel en HTML, CSS et JavaScript** : Le développement « en brut » consiste à créer le site intégralement à partir de langages web de base : HTML pour la structure, CSS pour la mise en forme, et JavaScript pour les interactions. Cette méthode implique de concevoir soi-même l'organisation des pages, la navigation, le design, l'intégration responsive pour les

mobiles, ainsi que les fonctionnalités interactives telles que les animations ou le formulaire de contact.

Cette approche demande davantage de temps et un niveau d'apprentissage plus élevé, car chaque élément doit être créé ou adapté manuellement. Elle offre cependant une maîtrise complète du fonctionnement du site et de son aspect visuel. Elle évite toute dépendance à un système externe, garantit un code léger et optimisé, et permet de personnaliser totalement le résultat. Le développement manuel est également l'occasion de mettre en pratique plusieurs compétences du BTS SIO SISR : compréhension des architectures web, déploiement sur un serveur Linux, gestion d'un domaine, sécurisation d'un formulaire, et configuration d'une solution d'hébergement.

2- Comparaisons des solutions

Critères	WordPress (CMS)	Solutions No-Code	Développement brut
Niveau de compétence requis	Moyen	Très faible	Élevé
Temps de mise en place	Rapide	Très rapide	Plus long
Hébergement	Serveur web nécessaire (LAMP)	Inclus dans la plateforme	Serveur web nécessaire (LAMP)
Personnalisation graphique	Élevée mais dépendante des thèmes	Limitée aux modèles proposés	Totale
Contrôle du code	Partiel	Aucun	Complet
Évolutivité	Bonne grâce aux plugins	Faible, dépend du service	Très bonne, extensible librement
Sécurité	Dépend de la maintenance des plugins et mises à jour	Gérée par la plateforme, peu maîtrisable	Dépend entièrement du développeur
Coûts	Basique gratuit, options premium possibles	Le plus souvent freemium (limitations sans abonnement)	Gratuit (hors hébergement et domaine)
Maintenance	Régulière (mises à jour thèmes/plugins)	Très faible (gérée par le fournisseur)	Moyenne (gestion du code et du serveur)
Performance	Variable (plugins parfois lourds)	Très bonne mais non optimisable	Optimisable au maximum
Respect des contraintes pédagogiques du BTS	Moyen (peu de manipulation technique)	Très faible (peu formateur)	Excellent (administration, déploiement, dev web)
Liberté et indépendance	Correcte mais dépend du CMS	Faible, dépend d'un service tiers	Totale

3- Choix d'une solution

Après comparaison des différentes approches possibles pour la création du site web, choix s'est porté sur la réalisation du site en développement "brut".

Cette solution se distingue par une liberté de création totale, permettant de concevoir une interface entièrement personnalisée, sans être contraint par les limites d'un thème ou d'un constructeur graphique. Contrairement aux plateformes no-code ou aux CMS, elle garantit également une indépendance complète vis-à-vis de services tiers, ce qui supprime les contraintes liées à un abonnement ou à l'évolution d'une plateforme.

Un autre avantage majeur réside dans la possibilité d'héberger le site soi-même, sur une infrastructure personnelle. Là où les solutions no-code imposent un hébergement externe et où WordPress nécessite souvent un hébergeur dédié, le développement en HTML/CSS/JS permet une maîtrise totale de l'environnement serveur. Cette autonomie permet de mobiliser pleinement les compétences acquises dans le cadre du BTS SIO SISR :

- installation d'un serveur Debian,
- mise en place d'une pile LAMP,
- gestion des services,
- configuration réseau et notions de sécurité,
- réflexion sur l'architecture (DMZ, pare-feu, filtrage, NAT...).

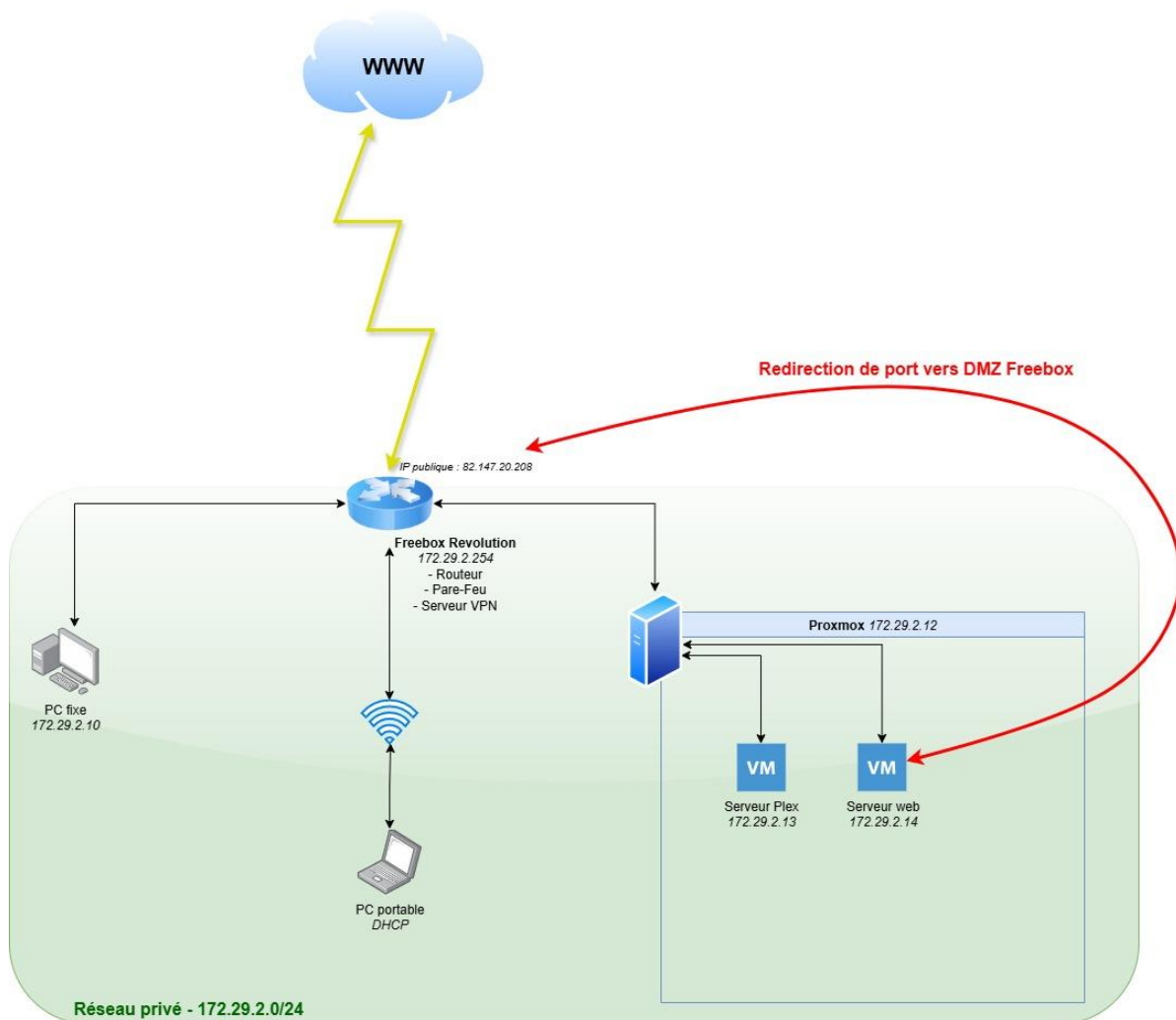
Sur le plan financier, le développement manuel est entièrement gratuit, mis à part le nom de domaine. L'inconvénient principal reste une durée de réalisation plus longue et un niveau de difficulté plus élevé, nécessitant des recherches régulières et une veille documentaire.

Cependant, ces contraintes s'inscrivent parfaitement dans une démarche pédagogique : elles renforcent l'apprentissage, permettent de mieux comprendre les mécanismes réels d'un site web et rendent le projet final plus abouti, plus maîtrisé et surtout plus satisfaisant.

En conséquence, la solution HTML/CSS/JavaScript s'avère la plus cohérente, la plus formatrice et la plus adaptée aux objectifs du projet.

4- Cartographie du réseau¹

a) Schéma



NB : La “DMZ” de la Freebox Révolution n’est pas une DMZ réseau au sens professionnel. Elle ne crée pas de zone isolée : elle se contente de rediriger l’ensemble du trafic entrant vers une machine précise du LAN. La VM hébergeant le site web devient donc l’hôte exposé, sans séparation du réseau local. Ce mécanisme reste fonctionnel pour un usage personnel, mais nécessite des mesures de sécurité locales (pare-feu, mises à jour, comptes limités, ports restreints).

Par sécurité j'envisage une DMZ via un pare-feu PFSense configuré en reverse-proxy afin de rediriger le trafic vers la DMZ qui le filtrera vers le serveur web sécurisant ainsi l'accès vers mon réseau privé.

b) Plan d'adressage

Réseau privé	172.29.2.0
Masque	255.255.255.0
1 ^{ère} adresse	172.29.2.1
Dernière adresse	172.29.2.254
Broadcast	172.29.2.255

¹ Toutes les adresses IP ont été modifiées et sont donc fictives

c) Tables de routage

- Freebox revolution :

Réseau	Adresse	Interface	Passerelle
Réseau privé	172.29.2.0/24	172.29.2.254	172.29.2.254
Default	0.0.0.0/0	82.147.20.208	FAI

5- Etude de l'impact sur le SI existant

La mise en place du site web personnel, hébergé sur une machine virtuelle dédiée au sein de l'infrastructure domestique, entraîne plusieurs impacts sur le système d'information existant. Ces effets concernent principalement la sécurité, la performance, les aspects juridiques et l'ergonomie. Une analyse préalable permet d'anticiper ces conséquences et d'adapter la solution de manière cohérente.

a) Sécurité

L'exposition d'un serveur web accessible depuis Internet constitue un point d'entrée potentiel pour des attaques externes (scans de ports, tentatives d'intrusion, exploitation de vulnérabilités). Afin de limiter les risques la freebox propose un pseudo DMZ permettant la redirection du trafic vers le serveur web.

Ces mesures réduisent significativement l'exposition aux risques, même dans le cadre d'un hébergement personnel.

b) Ergonomie

L'intégration d'un site web dans le SI personnel n'a pas d'impact direct sur l'ergonomie du réseau, mais elle implique des choix importants pour l'expérience utilisateur du site lui-même :

- Le site doit être accessible depuis n'importe quel appareil via un nom de domaine dédié.
- L'interface doit être lisible, intuitive et compatible avec les écrans mobiles.
- La structure du site doit permettre une navigation claire entre les différentes sections (profil, réalisations, procédures, veille, contact).
- L'utilisation d'un template existant, modifié et enrichi, assure une cohérence visuelle tout en permettant une personnalisation adaptée au contenu présenté.

Cette ergonomie contribue à faciliter l'accès et la compréhension du portfolio, autant pour les évaluateurs que pour de potentiels recruteurs.

6- Phasage de l'intervention

La création et la mise en production du site web nécessitent une organisation en plusieurs étapes successives, permettant d'assurer une progression logique, de réduire les risques d'erreur et de garantir un résultat fonctionnel et sécurisé. Le phasage ci-dessous détaille l'ordre chronologique des actions réalisées pour mener à bien ce projet.

• **Étape 1 — Préparation de l'environnement de travail**

La première phase consiste à mettre en place l'ensemble des outils nécessaires au développement. Cela inclut la création de l'architecture des dossiers du projet, ainsi que la mise en place d'un dépôt local afin de versionner les différentes évolutions. Cette étape prépare également l'environnement matériel, notamment le poste de travail personnel et l'accès aux ressources nécessaires (connexion Internet, documentation, alternance en entreprise).

- **Étape 2 — Conception et développement du site**

Le développement a débuté par l'analyse et l'adaptation du template HTML/CSS choisi comme base. Les pages principales ont été créées, puis enrichies progressivement : intégration du contenu, mise en place des animations JavaScript, création de la navigation, ajout des effets visuels et optimisation du rendu. Cette phase inclut également la mise en place du formulaire de contact et l'amélioration progressive de la structure du CSS afin de garantir une meilleure lisibilité et modularité du code.

- **Étape 3 — Mise en place du serveur et de l'hébergement**

Une fois l'environnement de développement opérationnel, l'installation du serveur web a été réalisée sur une machine virtuelle Debian hébergée sous Proxmox. L'installation de la pile LAMP a permis de fournir toutes les briques logicielles nécessaires au fonctionnement du site. Durant cette étape, le domaine a été réservé auprès d'OVH afin de permettre l'accès public au site, et les DNS ont été préparés pour pointer vers l'adresse IP publique du foyer.

- **Étape 4 — Mise en place de l'envoi des messages et sécurité**

Après la création du formulaire, l'envoi réel des messages a été mis en œuvre à l'aide de PHPMailer. Un fichier de configuration externe a été ajouté pour isoler les informations sensibles telles que les identifiants SMTP. Cette étape comprend aussi la configuration des permissions sur le serveur web, ainsi que les premières mesures de sécurité (sécurisation PHP, gestion des répertoires, vérifications côté serveur).

- **Étape 5 — Tests et corrections fonctionnelles**

Une fois le site globalement opérationnel, une série de tests fonctionnels a été réalisée : affichage des pages, fonctionnement du menu, compatibilité sur plusieurs navigateurs, tests mobiles, test du formulaire, validation des animations et contrôle de la cohérence visuelle. Les corrections nécessaires ont été apportées au fur et à mesure, notamment sur la partie JavaScript et CSS.

- **Étape 6 — Préparation de la mise en production**

Avant la mise en ligne définitive, l'arborescence du serveur a été vérifiée et les fichiers inutiles supprimés. Les enregistrements DNS ont été finalisés pour assurer la diffusion du domaine sur Internet. Une réflexion a été menée sur la sécurité réseau, notamment concernant la DMZ et le cloisonnement du serveur web, afin d'évaluer les pistes d'amélioration pour les déploiements futurs.

- **Étape 7 — Mise en production et suivi**

La dernière étape consiste à mettre le site en ligne, vérifier sa disponibilité depuis l'extérieur, contrôler les performances du serveur et s'assurer que le formulaire fonctionne correctement en conditions réelles. Un suivi continu est ensuite assuré pour corriger rapidement d'éventuels dysfonctionnements, améliorer l'ergonomie, enrichir le contenu et maintenir la sécurité du serveur au fil du temps.

7- Prévision des tests de validation

Avant la mise en production du site, une phase de tests est nécessaire afin de garantir son bon fonctionnement et sa conformité aux objectifs initiaux. Ces tests visent d'abord à s'assurer que l'ensemble des fonctionnalités essentielles est opérationnel, notamment la navigation générale, l'affichage des différentes pages et le comportement du menu mobile. Une attention particulière est

portée au formulaire de contact, dont la validation des champs, la transmission des données et la réception effective des courriels doivent être vérifiées en conditions réelles.

Sur le plan technique, il est indispensable de contrôler la stabilité du serveur web ainsi que la bonne configuration des services Apache, PHP et PHPMailer. Le système d'envoi SMTP doit être testé de manière à garantir la transmission sécurisée des messages. Les permissions des fichiers sensibles, notamment celles du fichier de configuration contenant les identifiants du serveur mail, sont également examinées, tout comme la résistance du formulaire aux erreurs de saisie ou aux tentatives malveillantes les plus courantes.

Le site doit ensuite être testé sur plusieurs types d'appareils afin de vérifier son adaptation aux différentes résolutions d'écran. L'affichage est contrôlé sur ordinateur, tablette et smartphone, ainsi que sur plusieurs navigateurs tels que Chrome, Firefox et Edge, afin d'écarter tout problème de compatibilité. Cette étape inclut également la surveillance des éventuelles erreurs remontées par la console du navigateur.

Enfin, des tests de performance et d'ergonomie permettent de mesurer le temps de chargement, d'évaluer la fluidité des animations et d'apprécier le confort de lecture dans un environnement graphique volontairement dynamique. Une fois ces vérifications effectuées, le site est testé depuis son nom de domaine OVH et depuis différents réseaux (fibre, Wi-Fi, 4G) afin de valider son accessibilité en conditions réelles.

8- Déploiement

Une fois les tests de validation réalisés et concluants, la phase de déploiement peut être engagée. Elle commence par la préparation de l'environnement serveur au sein de Proxmox, où la machine virtuelle dédiée au site web est finalisée avec Debian 13, Apache, PHP et l'ensemble des modules nécessaires au fonctionnement du formulaire de contact. Le fichier de configuration externe, contenant les identifiants SMTP, est placé de manière sécurisée et ses permissions sont ajustées afin de limiter les risques liés à un accès non autorisé.

Une fois le serveur opérationnel, le site est transféré dans son arborescence définitive. Les fichiers HTML, CSS, JavaScript, images et composants PHP sont intégrés au sein du répertoire destiné à l'hébergement. Après cette étape, une vérification est effectuée pour s'assurer que les dépendances installées via Composer, notamment PHPMailer, sont correctement détectées par le système.

Le déploiement se poursuit avec la configuration du domaine acquis chez OVH. Les enregistrements DNS sont ajustés de façon à pointer vers l'adresse publique du réseau domestique, tandis que les règles de redirection et de pare-feu sont mises à jour sur la Freebox afin de permettre l'accès au serveur web. Les ports nécessaires sont ouverts de manière contrôlée et documentée, en veillant à conserver un équilibre entre accessibilité et sécurité.

Après la mise en ligne, de nouveaux tests sont réalisés directement depuis l'URL publique afin de vérifier que l'ensemble du site reste fonctionnel en conditions de production. Cette étape essentielle permet de confirmer la bonne propagation DNS, la disponibilité du serveur depuis l'extérieur, ainsi que le bon acheminement des messages envoyés via le formulaire.

Le déploiement s'achève par une phase d'observation afin de détecter d'éventuels comportements anormaux, notamment des erreurs dans les logs Apache, des problèmes d'affichage ou des

ralentissements. Une fois la stabilité confirmée, le site est officiellement considéré comme opérationnel et accessible au public.

IV/ Mise en place

1- Réalisation

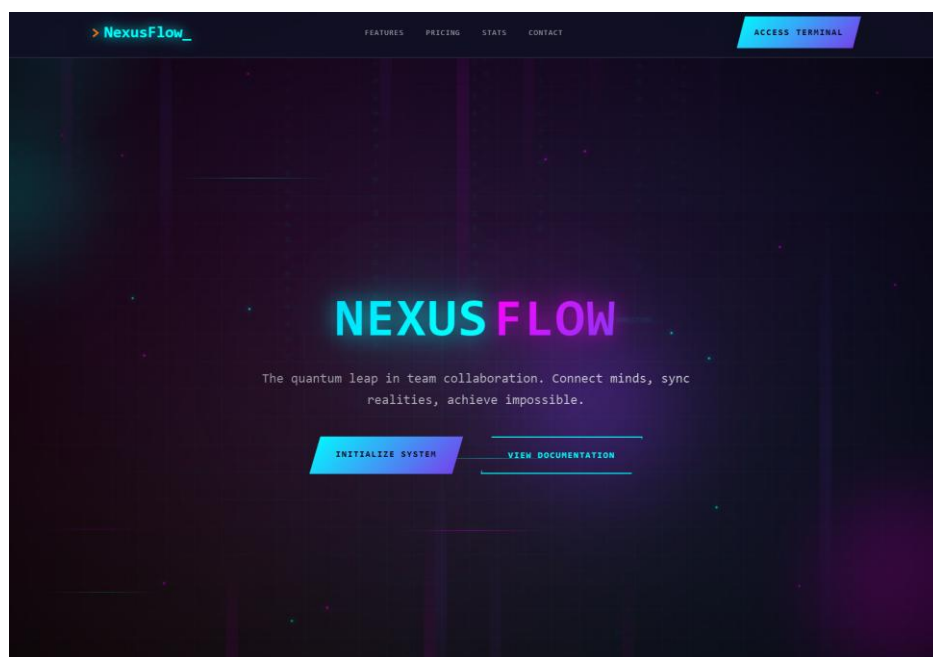
a) Recherche et choix d'un template

La première étape de la réalisation a consisté à rechercher un modèle visuel qui servirait de base à la construction du site. L'objectif était d'identifier une template moderne, esthétique et suffisamment flexible pour permettre une personnalisation complète tout en restant compatible avec une intégration manuelle en HTML, CSS et JavaScript.

Pour cela, plusieurs plateformes spécialisées dans les modèles de sites web ont été consultées, notamment TemplateMo, Tooplate, Nicepage ainsi que Figma, qui propose des maquettes graphiques plus orientées design que développement. Chacune de ces sources présente des approches différentes : certains modèles sont minimalistes, d'autres davantage orientés portfolio ou corporate, et quelques-uns incluent déjà des animations avancées.

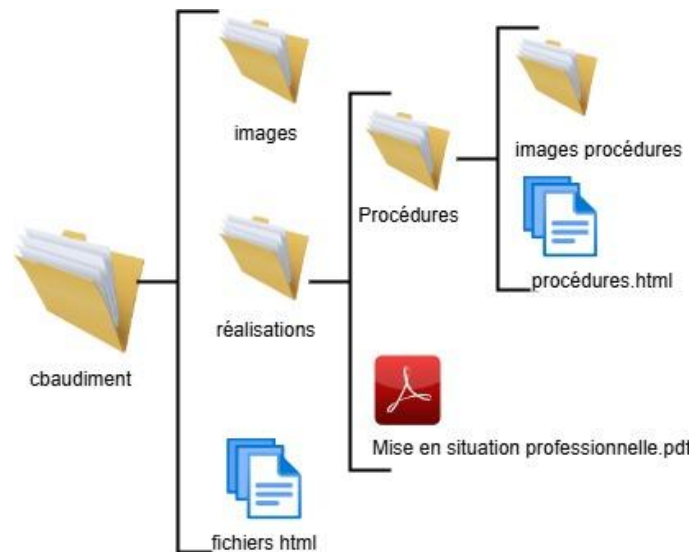
Après avoir comparé plusieurs propositions, la template Nexus Flow (TemplateMo n°594) a été retenue (<https://templatemo.com/tm-594-nexus-flow>). Son identité visuelle inspirée de l'univers cyberpunk, ses effets graphiques immersifs et son organisation claire en sections verticales correspondaient parfaitement au style recherché. Elle offrait un juste équilibre entre originalité, modernité et lisibilité, tout en permettant d'introduire une forte personnalisation pour coller aux besoins du projet.

Cette template constituera ainsi la base du travail de développement, à partir de laquelle les adaptations graphiques, la structure des pages et les différentes fonctionnalités seront progressivement intégrées.



b) Création du site

Avant de créer le site web j'ai créé le dossier cbaudiment où tous les éléments composant le site seront placer :



➤ Création du portfolio

c) Mise en place du serveur web et intégration du site

➤ Installation d'une VM sous Debian 13

➤ Configuration du réseau :

Une fois Debian installé nous allons le configurer et lui attribué une adresse IP fixe en allant dans son fichier de configuration réseau : `nano /etc/network/interfaces`

Le fichier sera configuré comme ci-dessous :

```
# The primary network interface
allow-hotplug ens18
iface ens18 inet static
    address 10.0.0.14
    netmask 255.255.255.0
    gateway 10.0.0.254
```

Pour être certains de sa connexion au réseau, je vais pouvoir le tester avec la commande `ping`.

```
root@portfolio:~# ping www.google.com
PING www.google.com (2a00:1450:4007:80b::2004) 56 data bytes
64 bytes from par10s40-in-x04.1e100.net (2a00:1450:4007:80b::2004) :
icmp_seq=1 ttl=112 time=10.3 ms
64 bytes from par10s40-in-x04.1e100.net (2a00:1450:4007:80b::2004) :
icmp_seq=2 ttl=112 time=9.35 ms
64 bytes from par10s40-in-x04.1e100.net (2a00:1450:4007:80b::2004) :
icmp_seq=3 ttl=112 time=9.56 ms
64 bytes from par10s40-in-x04.1e100.net (2a00:1450:4007:80b::2004) :
icmp_seq=4 ttl=112 time=9.45 ms
^C
--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 9.350/9.653/10.260/0.357 ms
root@portfolio:~#
```

➤ **Installation de la pile LAMP :**

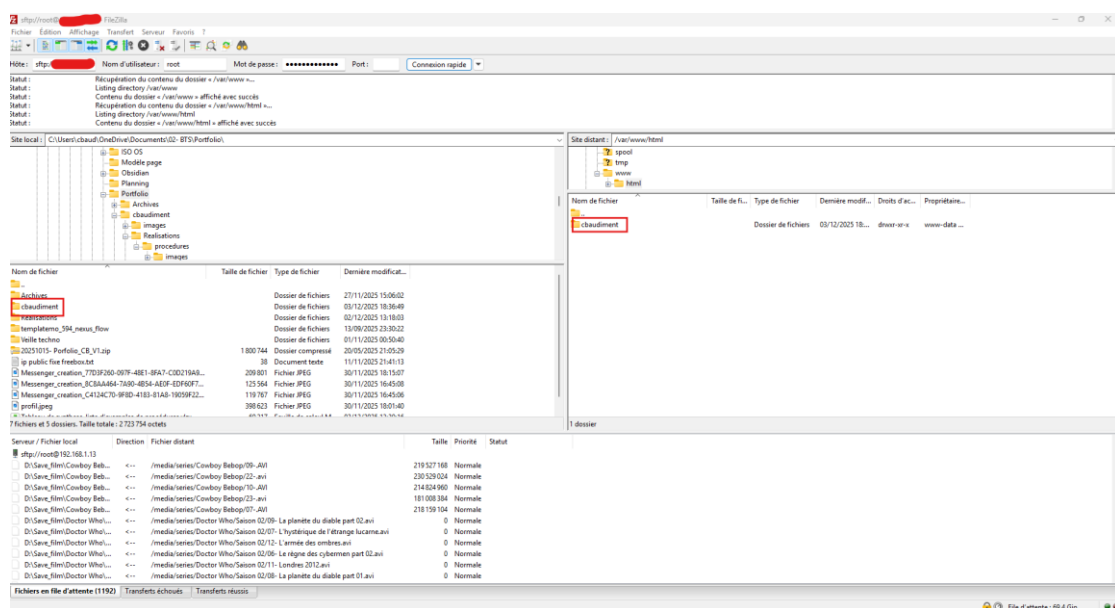
Pour que la machine virtuelle fonctionne correctement, je commence par mettre à jour le système avec la commande `apt update`. Une fois la mise à jour effectuée, j'installe les composants nécessaires à la pile LAMP, à savoir Apache2 pour le service web, MariaDB pour la base de données, ainsi que PHP et le module `php-mysql` permettant l'interfaçage avec MariaDB. Les commandes utilisées sont les suivantes :

- apt install apache2
- apt install mariadb-server
- apt install php
- apt install php-mysql

➤ **Installation du site web :**

Le dossier cbaudiment est placé dans le répertoire du serveur web à l'emplacement /var/www/html/cbaudiment.

Pour cela je vais utiliser le logiciel Filezilla pour transférer le dossier de mon PC fixe vers mon serveur web via le protocole sftp :



Pour que le site s'affiche automatiquement dès que l'on accède au domaine, je crée le fichier de configuration `cbaudiment.conf`, dont le contenu est

```
<VirtualHost *:80>
    ServerName portfolio-cbaudiment.fr
    ServerAlias www.portfolio-cbaudiment.fr

    DocumentRoot "/var/www/html/cbaudiment"
    <Directory "/var/www/html/cbaudiment">
        require all granted
        Options Indexes FollowSymLinks
        AllowOverride All
    </Directory>
</VirtualHost>
```

Ce fichier est déposé dans le répertoire /etc/apache2/sites-available. Il permet de rediriger automatiquement les requêtes adressées à portfolio-cbaudiment.fr ou www.portfolio-cbaudiment.fr vers le fichier index.html présent dans le dossier cbaudiment.

➤ Mise en place de la DMZ de la Freebox :

La Freebox Révolution propose une fonctionnalité appelée DMZ. Contrairement à une véritable DMZ, qui isole un serveur dans un réseau intermédiaire séparé du réseau local, la Freebox ne crée pas de zone dédiée. Elle se contente de rediriger l'ensemble du trafic entrant vers une adresse IP précise du réseau interne. Ainsi, le serveur déclaré en DMZ reste sur le même réseau local que les autres équipements, sans cloisonnement particulier

Pour la configuration, je commence par accéder à l'interface web de la Freebox depuis un navigateur. Je me rends ensuite dans Paramètres de la Freebox > Gestion des ports

J'active l'option DMZ puis j'indique l'adresse IP de mon serveur web afin que tout le flux extérieur soit redirigé vers cette machine.

Je configure ensuite une redirection du port 80 depuis l'adresse IP publique de la Freebox vers l'adresse IP locale du serveur web. Cette règle garantit que toute requête HTTP adressée à la Freebox aboutit directement sur le site hébergé par le serveur.

Liste des redirections				
Active	Redirection	IP source	Destination	
Active	Protocole: tcp WAN: 80 LAN: 80 Commentaire:		portfolio	 

DMZ

Activer la DMZ : ☒

IP DMZ :

 Rafraîchir  Ajouter une redirection

 OK  Annuler  Appliquer

d) Choix d'un nom de domaine sur OVH

Jusqu'à présent, l'accès au site se fait en saisissant l'adresse IP publique de la Freebox dans un navigateur. Pour pouvoir accéder au site via une adresse lisible, comme www.portfolio-cbaudiment.fr, il est nécessaire d'enregistrer un nom de domaine auprès d'un bureau d'enregistrement tel qu'OVHcloud. L'enregistrement est valable un an et peut être renouvelé annuellement.

Après la création de mon compte sur ovhcloud.com, je vérifie la disponibilité du domaine portfolio-cbaudiment.fr. Une fois la disponibilité confirmée, je procède à son enregistrement.

Une fois le domaine activé, je configure sa zone DNS en créant un enregistrement de type A pointant vers l'adresse IP publique de ma Freebox. Cette configuration permet de rediriger automatiquement les requêtes adressées au domaine vers mon serveur web.

portfolio-cbaudiment.fr
Renouvellement automatique prévu en nov. 2026 [Roadmap & i](#)

[Informations générales](#) **Zone DNS** [Serveurs DNS](#) [Redirection](#) [DynHost](#) [GLUE](#) [DS Records](#)

Vous pouvez voir ici la configuration des diverses entrées de votre domaine.
Vous avez également la possibilité de configurer ces entrées pour relier votre domaine à vos différents services (bouton « ajouter une entrée »).

Tous Recherche domaine... Q

☐ Domaine	TTL	Type	Cible	
☐ portfolio-cbaudiment.fr.	0	NS	dns16.ovh.net.	
☐ portfolio-cbaudiment.fr.	0	NS	ns16.ovh.net.	
☐ portfolio-cbaudiment.fr.	0	A		...
☐ www.portfolio-cbaudiment.fr.	0	A		...
☐ ftp.portfolio-cbaudiment.fr.	0	CNAME	portfolio-cbaudiment.fr.	...
☐ portfolio-cbaudiment.fr.	0	SPF	v=spf1 include:mx.ovh.com -all	...
☐ portfolio-cbaudiment.fr.	0	TXT	"1 www.portfolio-cbaudiment.fr"	...
☐ www.portfolio-cbaudiment.fr.	0	TXT	"3 welcome"	...
☐ portfolio-cbaudiment.fr.	0	MX	1 mx1.mail.ovh.net.	...
☐ portfolio-cbaudiment.fr.	0	MX	5 mx2.mail.ovh.net.	...

« < 1 2 3 > » 10 Page 1 / 3 OK

e) Envoi de message via la page contact

- [Procédure d'envoi de message via la page contact](#)

f) Adaptation du site aux supports mobiles

- [Configuration mobile responsive](#)

2- Rapport de tests

Avant la mise en production, l'ensemble du site web et de son infrastructure d'hébergement a été soumis à une série de tests fonctionnels, techniques et ergonomiques. Ces vérifications ont été réalisées progressivement au fil de l'avancement du développement, puis consolidées lors d'une phase finale de tests complète.

Les premières vérifications ont porté sur la structure du site, notamment la cohérence de la navigation et le comportement du menu, aussi bien sur ordinateur que sur appareils mobiles. Le fonctionnement du menu principal, du menu mobile et des liens internes a été entièrement contrôlé afin de s'assurer que chaque page était accessible sans erreur. Cette étape a permis de corriger plusieurs problèmes de responsivité, comme la position du bouton hamburger ou l'ajout de marges sur certaines pages dont le texte devenait trop large sur petit écran.

Une seconde série de tests a concerné l'affichage graphique et la cohérence du design sur l'ensemble du site. Les différents éléments visuels – effets de fond, animations, typographies, couleurs et

transitions – ont été analysés afin de garantir une expérience fluide et homogène. Des ajustements ont notamment été nécessaires pour harmoniser l'apparence des titres, améliorer la lisibilité sur mobile et optimiser l'affichage de la frise chronologique de la page Profil.

Le fonctionnement du formulaire de contact a fait l'objet d'une attention particulière, puisqu'il constitue un élément dynamique du site. Plusieurs tests d'envoi ont été effectués pour vérifier la bonne communication entre le site, le script PHP d'envoi et la configuration SMTP du serveur. La création du fichier config.php, la mise en place des bons identifiants SMTP OVH et la correction des permissions des fichiers ont permis d'assurer la réception correcte des messages dans la boîte mail personnelle. Chaque test a confirmé que les messages étaient envoyés instantanément.

Le serveur web a également été testé afin de vérifier sa stabilité et son accessibilité. L'affichage du site depuis différents appareils connectés au réseau local a permis de confirmer que la VM Debian répondait correctement aux requêtes HTTP. L'accès externe au site via le nom de domaine OVH a ensuite été vérifié, confirmant la bonne redirection DNS ainsi que la prise en compte de la configuration du serveur d'hébergement.

Enfin, une phase de test ergonomique a été effectuée pour vérifier que le site restait clair, intuitif et agréable à consulter pour un utilisateur non spécialiste. Les pages Profil, Formations, Réalisations et Contact ont été relues afin de vérifier la cohérence du contenu, la lisibilité et la fluidité de la navigation.

3- Rapport de déploiement

Le déploiement du site a débuté par la préparation de la machine virtuelle Debian 13 dédiée à l'hébergement. La pile LAMP a été installée, puis configurée pour accueillir un site statique en HTML/CSS/JS ainsi que PHPMailer pour la gestion du formulaire de contact. Les permissions des répertoires ont été ajustées afin de garantir à la fois sécurité et bon fonctionnement, notamment pour le dossier vendor généré par Composer.

Une fois le serveur opérationnel, l'ensemble des fichiers du site ont été transférés dans le répertoire web prévu à cet effet. Le fichier config.php, contenant les paramètres sensibles nécessaires à l'envoi d'e-mails, a été placé hors de la zone publique afin d'assurer sa protection. Apache a ensuite été configuré via un VirtualHost afin d'associer le domaine au répertoire du site, puis redémarré pour appliquer les modifications.

Le nom de domaine réservé sur OVH a été configuré pour pointer vers l'adresse IP publique du foyer, permettant l'accès externe au site après propagation DNS. Des vérifications ont ensuite été effectuées depuis différents appareils afin de s'assurer du bon chargement des pages, du rendu responsif et du fonctionnement du formulaire de contact.

L'ensemble des tests réalisés après déploiement a confirmé la stabilité du site et son accessibilité depuis l'extérieur, validant ainsi la mise en production finale.

v/ Bilan

1- Conclusion

La réalisation de ce site web s'est révélée être un projet formateur, à la croisée du développement web et de l'administration système. En partant d'un besoin initial simple — disposer d'un portfolio professionnel pour valoriser mes compétences et mes réalisations — le projet a progressivement

intégré des enjeux techniques concrets, notamment la mise en place d'un serveur web, la gestion de la sécurité, la configuration d'un nom de domaine et l'intégration d'un formulaire de contact fonctionnel et sécurisé.

Le choix d'une création en HTML, CSS et JavaScript s'est avéré pertinent, car il a offert une grande liberté de conception tout en renforçant ma compréhension des technologies web de base. L'hébergement autonome sur une infrastructure personnelle a également permis de mettre en pratique les compétences en réseau et en administration système développées au cours du BTS SIO SISR.

Le résultat final répond aux objectifs fixés : un site accessible, responsive, fonctionnel, esthétique et adapté à une utilisation professionnelle. Le projet a ainsi permis d'acquérir une vision complète du cycle de production d'un service web, depuis sa conception jusqu'à son déploiement effectif.

2- Auto-évaluation

Ce projet m'a permis de mesurer concrètement mes compétences actuelles ainsi que les points que je dois encore améliorer. J'ai su faire preuve de rigueur et d'autonomie dans la conception du site, la recherche d'informations, l'installation du serveur web et la résolution des problèmes rencontrés, notamment pour la configuration du formulaire de contact et la gestion de l'hébergement. J'ai également démontré ma capacité à apprendre rapidement, en particulier sur les aspects liés au développement web et à la sécurisation d'un service en ligne, domaines que je maîtrisais peu au départ.

Cependant, j'ai identifié plusieurs axes de progression, notamment l'organisation du code, la gestion du responsive design et l'optimisation des performances. Une meilleure anticipation de la structure globale du site aurait aussi permis de gagner du temps. Malgré ces points d'amélioration, ce projet a renforcé ma confiance en mes capacités et confirmé mon intérêt pour les métiers liés au réseau, à l'administration système et à la cybersécurité. Il constitue une étape importante dans ma reconversion et valide les compétences acquises durant cette première année de formation.