

BTS SIO - Option SISR

Lycée Théodore Aubanel - Avignon

Année scolaire 2025-2026

RAPPORT DE STAGE

Stage de 2ème année - 6 semaines

AMD COIFFURE

Création d'un Site Web Professionnel

<https://amdcoiffuresorgues.fr>

Réalisé par

Anis DAOUAIRI

Étudiant BTS SIO 2ème année - Option SISR

Janvier 2026

SOMMAIRE

1. Présentation du Projet	3
2. Les Étapes de Création	4
3. Architecture et Technologies	6
4. La Base de Données	8
5. Le Panneau d'Administration	10
6. La Sécurité du Site	12
7. Problèmes Rencontrés et Solutions	14
8. Vocabulaire Technique	16
9. Compétences Acquises	18
10. Conclusion	19
11. Annexes	20

1. Présentation du Projet

1.1 Le Client

AMD Coiffure est un salon de coiffure et barbier pour hommes situé à Sorgues (84700). Le gérant souhaitait avoir un site web professionnel pour :

- Présenter son salon et ses services sur internet
- Permettre aux clients de réserver en ligne 24h/24
- Recevoir des messages via un formulaire de contact
- Gérer facilement son activité via un panneau d'administration

Nom	AMD Coiffure - Barber Homme
Adresse	347 Avenue d'Avignon, 84700 Sorgues
Téléphone	07 81 36 61 20
Site web	https://amdcoiffuresorgues.fr

1.2 Objectif du Stage

Mon objectif pendant ce stage de 6 semaines était de créer un site web complet et fonctionnel, de A à Z. Cela comprend :

- La création des pages web (accueil, services, contact, réservation...)
- La mise en place d'une base de données pour stocker les informations
- Un système de réservation en ligne fonctionnel
- Un panneau d'administration sécurisé pour le gérant
- Le déploiement du site sur un hébergeur professionnel

2. Les Étapes de Création

Voici les grandes étapes que j'ai suivies pour créer le site, dans l'ordre chronologique :

2.1 Étape 1 : Analyse des Besoins

Avant de coder, j'ai d'abord discuté avec le client pour comprendre ses besoins :

- Quels services propose-t-il ? (coupes, barbes, dégradés...)
- Quels sont ses horaires d'ouverture ?
- Veut-il que les clients puissent réserver en ligne ?
- Quelles langues pour le site ? (Français, Anglais, Arabe)

2.2 Étape 2 : Création de la Maquette

J'ai ensuite créé une maquette du site pour visualiser le résultat final avant de coder. Cela permet de valider le design avec le client.

2.3 Étape 3 : Développement du Front-end

Le front-end, c'est la partie visible du site (ce que voit l'utilisateur). J'ai créé :

- La structure des pages en HTML (le squelette)
- Le design et les couleurs en CSS (l'habillage)
- Les animations et interactions en JavaScript (le dynamisme)

2.4 Étape 4 : Développement du Back-end

Le back-end, c'est la partie invisible qui gère les données. J'ai créé :

- La base de données MySQL pour stocker les informations
- Les scripts PHP pour traiter les formulaires
- Les API pour la réservation et le contact
- Le panneau d'administration protégé par mot de passe

2.5 Étape 5 : Tests et Corrections

J'ai testé toutes les fonctionnalités pour m'assurer qu'elles fonctionnent correctement :

- Test de la réservation en ligne
- Test du formulaire de contact
- Test de la connexion admin
- Test sur mobile et tablette (responsive)

2.6 Étape 6 : Mise en Ligne

Enfin, j'ai déployé le site sur l'hébergeur Hostinger :

- Achat du nom de domaine : amdcoiffuresorgues.fr
- Configuration de l'hébergement
- Upload des fichiers via FTP
- Activation du certificat SSL (HTTPS)

3. Architecture et Technologies

3.1 Les Technologies Utilisées

J'ai utilisé plusieurs technologies pour créer ce site. Voici un tableau récapitulatif avec des explications simples :

Technologie	Rôle	Explication Simple
HTML	Structure	C'est le squelette de la page : titres, textes, images, boutons
CSS	Design	C'est l'habillage : couleurs, polices, espacements, animations
JavaScript	Interactivité	C'est le cerveau : réactions aux clics, formulaires dynamiques
PHP	Traitement	C'est le moteur côté serveur : il traite les données et parle à la BDD
MySQL	Stockage	C'est le coffre-fort : il stocke clients, RDV, messages, etc.
PDO	Connexion BDD	C'est le traducteur entre PHP et MySQL (sécurisé)

3.2 Organisation des Fichiers

Le site est organisé en dossiers pour que ce soit clair et facile à maintenir :

Dossier	Contenu
/	Pages principales : index.php, services.php, contact.php, reservation.php
/admin/	Panneau d'administration (protégé par mot de passe)
/api/	Scripts qui traitent les données (booking.php, contact.php)
/assets/css/	Fichiers de style (design du site)
/assets/js/	Fichiers JavaScript (interactions)
/includes/	Fichiers PHP partagés : config.php, Database.php, fonctions.php

3.3 Comment ça Fonctionne ?

Voici le parcours d'une réservation, étape par étape :

1. Le client va sur la page reservation.php
2. Il choisit un service, une date et un créneau horaire
3. Il remplit ses coordonnées (nom, téléphone, email)
4. JavaScript envoie les données à l'API (api/booking.php)
5. PHP vérifie les données et les enregistre dans MySQL
6. Le client voit un message de confirmation

7. Le gérant voit le RDV dans son panneau admin

4. La Base de Données

4.1 Qu'est-ce qu'une Base de Données ?

Une base de données, c'est comme un grand classeur numérique avec plusieurs tiroirs (les tables). Chaque tiroir contient des fiches (les lignes) avec des informations organisées en colonnes.

Par exemple, le tiroir 'clients' contient toutes les fiches clients avec leur nom, téléphone, email, etc.

4.2 Les Tables du Site AMD Coiffure

J'ai créé 10 tables pour gérer toutes les données du site :

Table	À quoi ça sert ?
admins	Stocke les comptes administrateurs (identifiant + mot de passe crypté)
appointments	Stocke tous les rendez-vous (date, heure, client, service)
clients	Stocke les informations des clients (nom, téléphone, email)
services	Liste des prestations (coupe, barbe, dégradé...) avec les prix
opening_hours	Horaires d'ouverture pour chaque jour de la semaine
messages	Messages reçus via le formulaire de contact
gallery	Photos de la galerie (réalisations du salon)
reviews	Avis des clients avec notes et commentaires
settings	Paramètres généraux du site
statistics	Statistiques de visites du site

4.3 Exemple : La Table 'appointments'

Voici à quoi ressemble la table des rendez-vous :

Colonne	Type	Description
id	INT (auto)	Numéro unique du RDV (1, 2, 3...)
client_name	VARCHAR(100)	Nom du client
client_phone	VARCHAR(20)	Téléphone du client
service_id	INT	Numéro du service choisi
appointment_date	DATE	Date du RDV (ex: 2026-02-15)
appointment_time	TIME	Heure du RDV (ex: 14:30)
status	ENUM	Statut : pending, confirmed, completed, cancelled

4.4 La Connexion avec PDO

Pour connecter PHP à la base de données MySQL, j'utilise PDO (PHP Data Objects). C'est une méthode sécurisée qui protège contre les injections SQL.

J'ai créé une classe Database.php qui gère la connexion :

- Elle utilise le pattern Singleton (une seule connexion partagée)
- Elle propose des méthodes simples : fetch(), fetchAll(), insert(), update(), delete()
- Elle utilise des requêtes préparées pour la sécurité

5. Le Panneau d'Administration

5.1 À Quoi Sert le Panneau Admin ?

Le panneau d'administration permet au gérant de gérer son site sans toucher au code. C'est une interface protégée par mot de passe accessible uniquement par lui.

5.2 Fonctionnalités Disponibles

Page Admin	Ce qu'on peut faire
Tableau de bord	Voir les statistiques : nombre de RDV, clients, messages
Rendez-vous	Voir, confirmer, annuler ou marquer comme terminé les RDV
Clients	Voir la liste des clients et leurs informations
Messages	Lire les messages du formulaire de contact
Galerie	Ajouter ou supprimer des photos
Services	Modifier les prestations et les tarifs
Horaires	Modifier les heures d'ouverture
Avis	Valider ou supprimer les avis clients

5.3 Accès au Panneau Admin

URL	https://amdcoiffuresorgues.fr/admin/
Identifiant	admin
Mot de passe	Amd2024!Secure

5.4 Comment Fonctionne la Connexion Admin ?

Quand le gérant se connecte, voici ce qui se passe :

8. Il entre son identifiant et mot de passe sur la page login.php
9. PHP vérifie si l'identifiant existe dans la table 'admins'
10. PHP compare le mot de passe avec le hash stocké (password_verify)
11. Si c'est correct, PHP crée une session sécurisée
12. Le gérant est redirigé vers le tableau de bord
13. Chaque page admin vérifie si la session existe, sinon retour au login

6. La Sécurité du Site

La sécurité est très importante pour un site web. Voici toutes les mesures que j'ai mises en place :

6.1 Protection contre les Injections SQL

L'injection SQL, c'est quand un pirate essaie d'entrer du code malveillant dans un formulaire pour accéder à la base de données.

Exemple d'attaque : entrer ' OR '1'='1 dans un champ pour contourner un mot de passe.

Ma protection : j'utilise des requêtes préparées avec PDO. Les données sont traitées comme du texte, jamais comme du code.

Exemple de code sécurisé :

```
$stmt = $pdo->prepare("SELECT * FROM users WHERE email = :email");
$stmt->execute(['email' => $email]);
```

Le :email est un paramètre nommé. PDO l'échappe automatiquement, donc même si quelqu'un entre du code malveillant, il sera traité comme du simple texte.

6.2 Protection des Mots de Passe

Les mots de passe ne sont jamais stockés en clair dans la base de données. J'utilise le hachage avec bcrypt.

Comment ça marche :

- password_hash('monMotDePasse', PASSWORD_DEFAULT) → crée un hash unique
- password_verify('monMotDePasse', \$hash) → vérifie si le mot de passe correspond

Même si un pirate accède à la base de données, il ne verra que des hash illisibles, pas les vrais mots de passe.

6.3 Certificat SSL (HTTPS)

Le site utilise HTTPS au lieu de HTTP. Cela signifie que toutes les données échangées entre le visiteur et le serveur sont cryptées.

Avantages :

- Les mots de passe ne peuvent pas être interceptés
- Les données personnelles sont protégées
- Google favorise les sites HTTPS dans les résultats de recherche
- Le cadenas vert rassure les visiteurs

6.4 Protection des Sessions

Quand un admin se connecte, PHP crée une session sécurisée :

- Un identifiant unique est généré (session_id)
- Cet ID est stocké dans un cookie sécurisé
- À chaque page, PHP vérifie si la session est valide
- La session expire après 30 minutes d'inactivité

6.5 Validation des Données

Toutes les données entrées par les utilisateurs sont vérifiées :

- Côté client (JavaScript) : vérification immédiate dans le navigateur
- Côté serveur (PHP) : re-vérification obligatoire avant traitement

On ne fait jamais confiance aux données côté client car elles peuvent être modifiées.

6.6 Tableau Récapitulatif de la Sécurité

Menace	Protection	Comment
Injection SQL	Requêtes préparées	PDO avec paramètres nommés
Vol de mot de passe	Hachage bcrypt	password_hash() / password_verify()
Interception données	HTTPS / SSL	Certificat Let's Encrypt
Accès non autorisé	Sessions PHP	Vérification sur chaque page admin
Données invalides	Double validation	JS (client) + PHP (serveur)

7. Problèmes Rencontrés et Solutions

Pendant le développement, j'ai rencontré plusieurs problèmes techniques. Voici comment je les ai résolus :

7.1 Erreur PDO : Paramètres Mixtes

Problème	Message d'erreur : 'mixed named and positional parameters'
Cause	Je mélangeais les ? et les :nom dans mes requêtes SQL
Solution	J'ai remplacé tous les ? par des paramètres nommés (:email, :phone...)

7.2 Conflit de Fonctions PHP

Problème	Message d'erreur : 'Cannot redeclare getServices()'
Cause	J'avais une fonction getServices() dans deux fichiers différents
Solution	J'ai renommé les fonctions de l'API avec le préfixe 'api' (apiGetServices)

7.3 Formulaire Contact : Champs Vides

Problème	Le formulaire disait 'champs obligatoires' même quand tout était rempli
Cause	Un autre script (main.js) vidait les champs avant l'envoi
Solution	J'ai changé le bouton de type='submit' à type='button' avec onclick

7.4 Créneaux Non Affichés

Problème	'Erreur de chargement' quand je choisissais une date pour réserver
Cause	La table 'opening_hours' était vide dans la base de données
Solution	J'ai créé un script d'installation pour remplir les horaires par défaut

7.5 Problème de Cache Navigateur

Parfois, mes modifications n'apparaissaient pas car le navigateur gardait l'ancienne version en mémoire.

Solution : vider le cache avec Ctrl+Shift+Suppr ou utiliser la navigation privée pour tester.

8. Vocabulaire Technique

Voici les termes techniques utilisés dans ce projet avec des explications simples :

Terme	Définition Simple
API	Interface de Programmation. C'est un 'traducteur' qui permet à deux programmes de communiquer. Ici, JavaScript parle à PHP via l'API.
AJAX	Technique pour envoyer des données au serveur sans recharger la page. C'est ce qui rend le site fluide.
Back-end	La partie invisible du site : serveur, base de données, traitement des données. C'est la cuisine du restaurant.
BDD / Base de données	Un grand classeur numérique qui stocke toutes les informations de manière organisée.
CSS	Langage qui définit le design : couleurs, polices, espacements, animations.
DNS	Système qui traduit les noms de domaine en adresses IP. Il transforme amdcoiffuresorgues.fr en 123.45.67.89.
Front-end	La partie visible du site : ce que l'utilisateur voit et avec quoi il interagit. C'est la salle du restaurant.
FTP	Protocole pour transférer des fichiers vers le serveur. J'utilise FileZilla pour ça.
Hash / Hachage	Transformer un mot de passe en code illisible. On ne peut pas revenir en arrière.
HTML	Langage qui structure le contenu : titres, paragraphes, images, liens...
HTTPS / SSL	Version sécurisée de HTTP. Les données sont cryptées pendant le transport.
Injection SQL	Attaque où le pirate entre du code SQL dans un formulaire pour voler des données.
JavaScript	Langage qui rend le site interactif : réactions aux clics, animations, formulaires dynamiques.
JSON	Format de données léger pour échanger des informations. Exemple : <code>{'nom': 'Ali', 'age': 25}</code>
MySQL	Système de base de données. C'est le logiciel qui gère le stockage des données.
PDO	PHP Data Objects. Interface sécurisée pour que PHP communique avec MySQL.
PHP	Langage côté serveur qui traite les données et génère les pages web.
Responsive	Un site qui s'adapte à toutes les tailles d'écran : PC, tablette, mobile.
Session	Mécanisme pour 'se souvenir' d'un utilisateur connecté pendant sa visite.
Singleton	Pattern de programmation : une seule instance d'une classe. Ici, une seule connexion BDD.

9. Compétences Acquises

9.1 Compétences Techniques

Domaine	Ce que j'ai appris
HTML / CSS	Créer des pages web modernes et responsive
JavaScript	Rendre un site interactif, faire des appels API (AJAX/fetch)
PHP	Créer des scripts serveur, gérer les sessions, traiter les formulaires
MySQL	Créer et gérer une base de données, écrire des requêtes SQL
Sécurité	Protéger contre les injections SQL, hacher les mots de passe, utiliser HTTPS
Déploiement	Mettre en ligne un site, configurer un hébergement, utiliser FTP
Debugging	Trouver et corriger les erreurs avec la console navigateur

9.2 Compétences Transversales

- Autonomie : j'ai travaillé seul sur ce projet de bout en bout
- Résolution de problèmes : j'ai appris à chercher des solutions face aux erreurs
- Communication : j'ai su comprendre les besoins du client
- Organisation : j'ai structuré mon travail et respecté les délais
- Documentation : j'ai appris à documenter mon travail pour le transmettre

10. Conclusion

10.1 Bilan du Projet

Ce stage de 6 semaines a été une expérience très enrichissante. J'ai pu réaliser un projet complet, de l'analyse des besoins jusqu'à la mise en ligne du site.

Le site AMD Coiffure est maintenant fonctionnel et utilisé par le gérant au quotidien. Les clients peuvent réserver en ligne 24h/24, ce qui facilite la gestion du salon.

10.2 Ce que j'ai le Plus Appris

Voici les trois apprentissages les plus importants de ce stage :

14. La sécurité n'est pas optionnelle : protéger les données des utilisateurs est une responsabilité importante.
15. Le debugging fait partie du métier : savoir lire une erreur et la résoudre est une compétence essentielle.
16. La communication avec le client est clé : comprendre ses besoins réels évite de perdre du temps.

10.3 Améliorations Futures

Si je devais continuer ce projet, voici ce que j'ajouterais :

- Envoi d'emails automatiques pour confirmer les réservations
- Rappels SMS la veille du rendez-vous
- Système de paiement en ligne
- Application mobile pour les clients

10.4 Remerciements

Je tiens à remercier le gérant d'AMD Coiffure pour sa confiance et sa disponibilité. Ce projet m'a permis de mettre en pratique mes connaissances et d'en acquérir de nouvelles dans un contexte professionnel réel.

11. Annexes

11.1 Informations du Site

Site public	https://amdcoiffuresorgues.fr
Admin URL	https://amdcoiffuresorgues.fr/admin/
Identifiant admin	admin
Mot de passe admin	Amd2024!Secure

11.2 Hébergement

Hébergeur	Hostinger
Type	Hébergement mutualisé
Base de données	u407289818_amd_coiffure
SSL	Let's Encrypt (HTTPS)

11.3 Outils Utilisés

- Visual Studio Code : éditeur de code
- FileZilla : transfert de fichiers FTP
- phpMyAdmin : gestion de la base de données
- Chrome DevTools : debugging et tests
- Git : versionnage du code (en local)

— *Fin du document* —

Anis DAOUAIRI - BTS SIO SISR - Stage 2ème année - Janvier 2026