

Guide Technique - Projet: Mon Salon Coiffure

Technologies utilisées

Frontend : Angular, HTML, CSS, Bootstrap
Backend : PHP (API REST)
Base de données : MySQL
Librairies : FullCalendar, ngx-bootstrap
Hébergement : Infomaniak (frontend, backend, base de données)
Outils de développement : Php Storms, XAMPP, GitHub

Architecture du projet

Mon-salon-coiffure/
├── frontend/ → Application Angular (interface client et admin)
├── backend/ → API PHP (gestion utilisateurs + réservations)
├── database.sql → Script SQL de création de la base
└── README.md → Documentation du projet

Installation locale (environnement de test)

1 Cloner le dépôt :

```
git clone https://github.com/ciss5/Projet_Fin_Etude.git
```

2 Installer les dépendances Angular :

```
cd frontend  
npm install
```

3 Lancer Angular :

```
ng serve → http://localhost:4200
```

4 Lancer le backend PHP :

```
cd backend  
php -S localhost:8000 → http://localhost:8000
```

5 Importer la base de données :

Importer database.sql dans phpMyAdmin ou via MySQL CLI

🌐 Accès en ligne (hébergement Infomaniak)

Site Angular : <https://beautyglow.ciss-mame.fr/>

API Backend : <https://beautyglow.ciss-mame.fr/backend/controllers/reservation.php>

Base de données : hébergée chez Infomaniak

⌚ Connexion à la base de données (Infomaniak)

- Hôte : 276kvu.myd.infomaniak.com
- Base de données : 276kvu_beautyglow_db
- Utilisateur : 276kvu_beauty
- Mot de passe : 2412BeautyGlow_user@

Fichier concerné : backend/config/config.php

🔒 Comptes de test

Administrateur :

Email : admin@coiffure.com

Mot de passe : admin123

Utilisateur :

Email : user@test.com

Mot de passe : test123

📁 Structure du code (backend PHP)

users.php : Inscription et connexion

reservation.php : CRUD des réservations

config/config.php : Connexion MySQL sécurisée
mail.php : Notifications par mail

Structure du code (frontend Angular)

services/ : Communication avec l'API (auth, réservation)
guards/admin.guard.ts : Sécurisation des routes admin
components/ : Composants Angular (login, calendrier, admin)
app-routing.module.ts : Routing Angular

Sécurité mise en place

- Mots de passe hachés (PHP)
- Validation côté client et serveur
- Requêtes préparées (protection SQL)
- AdminGuard pour restreindre les routes
- Pas de réservation le lundi / horaires hors plage

Fonctionnalités principales

Utilisateurs :

- Créer un compte, se connecter
- Réserver un créneau
- Recevoir un mail de confirmation

Administrateur :

- Se connecter
- Gérer les réservations (accepter / refuser)
- Notification automatique par mail