

1. Contexte et Objectif

Dans un monde où la sécurité des informations est primordiale, une agence d'espionnage fictive a besoin d'un **outil de communication sécurisé** pour ses agents. Le but de ce projet est de concevoir un site web permettant aux espions de :

1. **S'authentifier** grâce à des identifiants sécurisés.
2. **Communiquer via un chat interne** tout en chiffrant leurs messages pour garantir la confidentialité.

Les agents utiliseront un **chiffrement par le code César**, un système simple mais efficace pour cacher les messages des regards indiscrets.

Mission confiée aux étudiants

Vous êtes une équipe de développeurs en formation, engagée pour réaliser cet outil. L'objectif est de livrer un prototype fonctionnel qui garantit :

- **La sécurité des accès (authentification sécurisée).**
- **Le chiffrement des messages.**
- **Une gestion simplifiée des données utilisateurs.**

Thème global : Vous travaillez pour une agence appelée "**ShadowComm**", spécialisée dans la communication cryptée pour les espions infiltrés.

2. Fonctionnalités Principales

2.1. Accès Sécurisé à l'Agence

- **Inscription des agents :**
 - Les agents doivent créer un compte avec un nom de code et un mot de passe.
 - Avant de s'inscrire, ils doivent cocher une case pour accepter les règles de confidentialité de l'agence.
 - **Connexion des agents :**
 - Chaque agent doit saisir son nom de code et son mot de passe pour accéder au chat.
 - Une fois connecté, il pourra interagir avec d'autres espions en ligne.
 - **Sécurité :**
 - Les mots de passe doivent être hashés pour éviter les fuites d'informations sensibles en cas de compromission de la base de données.
-

2.2. Communication Sécurisée via le Chat

- Une fois connectés, les espions accèdent à une **interface de chat en temps réel**.

- Chaque message envoyé est :
 - **Chiffré avec le code César** avant d'être stocké dans la base de données.
 - **Déchiffré** automatiquement lorsqu'il est affiché pour l'utilisateur.
 - L'interface doit afficher :
 - Le nom de code de l'expéditeur.
 - Le contenu déchiffré du message.
 - L'heure d'envoi.
-

2.3. Gestion des Missions et Données Utilisateurs

- **Base de données :**
 - Une table users pour stocker les agents :
 - id, username, password.
 - Une table messages pour stocker les messages échangés :
 - id, sender_id, content (chiffré), timestamp.
 - **Suppression d'un agent :**
 - Un espion peut demander à l'agence de supprimer toutes ses données (RGPD simplifié).
 - Cela doit :
 - Supprimer son compte.
 - Effacer tous ses messages de la base.
-

3. Exigences Techniques

3.1. Frontend (HTML/CSS)

- **Thème visuel :**
 - Design sombre et épuré, pour coller à l'ambiance espionnage.
 - Exemple : arrière-plan noir, éléments aux teintes grises ou rouges, typographie futuriste.
- **Pages nécessaires :**
 - **Accueil** : Présentation de l'agence ShadowComm (fictive) et formulaire de connexion.
 - **Inscription** : Formulaire pour créer un compte d'espion.

- **Interface de chat** : Une zone de messages et un champ pour écrire/envoyer des messages.
-

3.2. Backend (PHP)

- Gestion des fonctionnalités :
 - **Inscription et Connexion sécurisées** :
 - Validation des données (nom unique, mot de passe requis).
 - Utilisation de `password_hash()` et `password_verify()` pour gérer les mots de passe.
 - **Messages** :
 - Chiffrement et déchiffrement via une fonction PHP personnalisée pour le code César.
 - **Gestion des requêtes SQL** :
 - Utilisation de PDO pour prévenir les injections SQL.
-

3.3. Chiffrement par le Code César

- **Règles** :
 - Décalage fixe des lettres de l'alphabet (par exemple : clé = 3).
 - Chiffrement des messages avant stockage.
 - Déchiffrement avant affichage.

Exemple :

- Message clair : "RENDEZ-VOUS À MINUIT".
 - Message chiffré : "UHQGHJ-YRXV É PLQXLW".
-

3.4. Collaboration avec Git

- **Dépôt Git** :
 - Chaque groupe devra héberger son code sur un dépôt Git.
- **Organisation des branches** :
 - Branches séparées pour les différentes fonctionnalités (authentification, chat, chiffrement).
- **Commits clairs** :
 - Exemple : "Ajout de la fonction de chiffrement César."

5. Résultats Attendus

- Un site fonctionnel avec une ambiance immersive d'espionnage.
- Une gestion sécurisée des utilisateurs et de leurs données.
- Un chat opérationnel avec chiffrement César.
- Un dépôt Git structuré, reflétant un bon travail d'équipe.