

Cahier des Charges Fonctionnel *(CdCF)*

Lucas CHAPRON - e2105151
Bastien COUTAND - e2100676
Robin MARCHAND - e2101234

4A - S8
2022 / 2023

Projet encadré par

Salah SADOU

Représentant du client

Maykel MATTAR

Table des matières

1	Cadre du projet
1.1	Résumé
1.2	Enjeux et objectifs
1.3	Livrables
1.4	Présentation de l'équipe
1.5	Planning prévisionnel
2	Spécifications fonctionnelles
2.1	Périmètre fonctionnel
2.1.1	Front office
2.1.2	Back office
2.3	Aperçu des contenus
3	Spécifications techniques
3.1	Choix technologiques
3.2	Accessibilité.....
3.2.1	Compatibilité navigateurs
3.2.2	Types d'appareils
3.3	Sécurité

1 Cadre du projet

1.1 Résumé

Le commanditaire du projet M. Mattar, faisant parti de l'entreprise DAWIZZ et chercheur à l'ERISA, a exprimé les besoins suivants :

- ↗ Mise en place d'une plateforme web permettant d'obfusquer et de désobfusquer des images et pdf.

1.2 Enjeux et objectifs

Objectif : Créer une interface WEB permettant de cacher et récupérer des informations d'une image ou d'un pdf.

Enjeux : Une fois qu'un pdf/une image est mis(e) en ligne, il n'y a plus aucune gestion d'accès puisque toutes les informations sont visibles. Seulement, certaines données sont dites sensibles tel qu'un numéro de téléphone ou une adresse. Il faut donc une solution pour pouvoir cacher et obtenir ces informations si on en a les droits.

1.3 Livrables

- ↗ Fichiers sources ainsi que fichiers de déploiement (le déploiement doit être simplifié par l'usage de container)
- ↗ Documentation utilisateur

1.4 Présentation de l'équipe

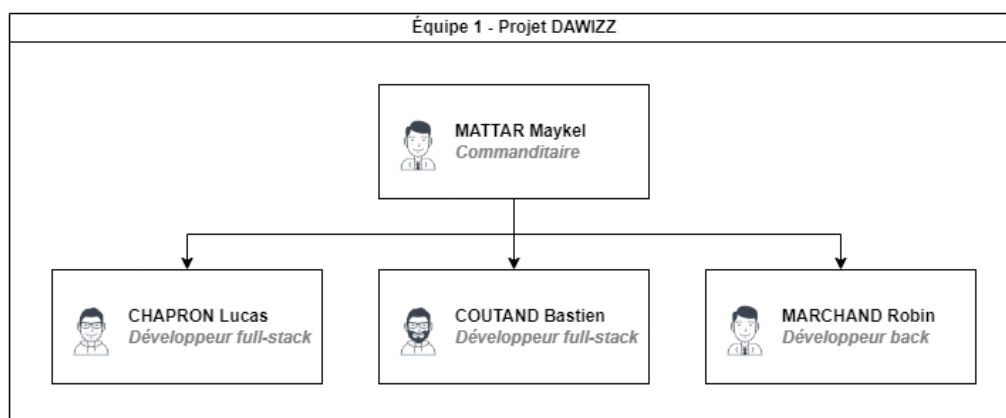


Figure 1 – Diagramme hiérarchique de l'équipe de développement

1.5 Planning prévisionnel

Présentation du cahier des charges	semaine 7
Soutenance présentation application	semaine 11
Soutenance final	semaine 16

Pour voir le gantt du projet voir [ProjetS8_Gantt_Init.html](#) joint.

2 Spécifications fonctionnelles

2.1 Périmètre fonctionnel

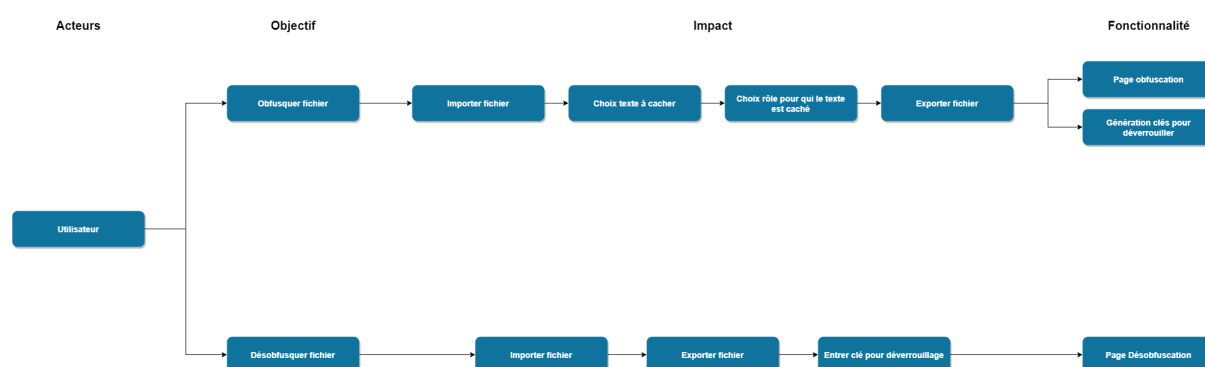


Figure 2 – Diagramme fonctionnel de l'application web

2.1.1 Front office

2.1.1.1 Page d'obfuscation

Fonctionnalité	Contrainte(s) associée(s)
Importer fichier	Uniquement png et pdf
Choix texte à cacher	OCR pour du texte
Choix rôle pour qui le texte est caché	-
Exporter fichier	Uniquement png

2.1.1.2 Page de désobfuscation

Fonctionnalité	Contrainte(s) associée(s)
Importer fichier	Uniquement png
Entrer clé pour déverrouillage	Clé valide
Exporter fichier	Uniquement png

2.1.2 Back office

2.1.2.1 Extraction du texte d'une image

Objectif :

- ↗ Extraire le texte d'une image avec de l'OCR et le rendre lisible pour le WEB

Contraintes :

- ↗ Extraction du texte utile uniquement

Exemple :

Dans la page pour obfusquer un fichier, un CV est importé pour cacher le nom, prénom, le numéro de téléphone et l'email de la personne. Pour éviter d'avoir une liste de texte sélectionnable importante, on ne souhaite pas pouvoir cacher des informations comme sa formation ou son expérience uniquement celles dites sensibles.

2.1.2.2 Chiffrer les données à cacher

Objectif :

- ↗ Chiffrer le texte, pour que même si un attaquant arrive à extraire le texte il ne puisse pas le lire et ne puisse pas bruteforcer les clés/casser l'algorithme.

Contraintes :

- ↗ Utiliser une clé suffisamment grande pour ne pas être cassable.

2.1.2.3 Cacher dans l'image le texte obfusqué

Objectif :

- ↗ Chiffrer le texte, pour que même si un attaquant arrive à extraire le texte il ne puisse pas le lire et ne puisse pas bruteforcer les clés/casser l'algorithme.

Contraintes :

- ↗ La modification de l'image ne doit pas être visible pour ne pas révéler les pixels modifiés.

Exemple :

La modification des bits de poids faible (LSB) consiste à modifier des pixels d'une image, si on modifie de trop la valeur RGB de celle-ci, avec des techniques de stéganographie classique on voit très vite que des pixels ont été modifiés.

2.1.2.4 Récupérer le texte en fonction de la clé de déchiffrement

Objectif :

- ↗ Avec la clé de déchiffrement, on doit être capable de récupérer le texte chiffré dans l'image et de le déchiffrer.

Contraintes :

- ↗ La clé doit permettre de récupérer la position du texte modifié.

2.3 Aperçu des contenus

Désobfuscation

Obfuscation

Désobfuscation fichier (PDF, png, jpg seulement)

Importer

Nom_du_fichier.extension

Télécharger

String pour unlock en fonction du rôle

DAWIZZ

Figure 3 – Page désobfuscation

Désobfuscation

Obfuscation

Obfuscation fichier (PDF, png, jpg seulement)

Importer

Nom_du_fichier.extension

Télécharger

Plusieurs sélectionnable



21 ans

Lucas Chapron

Rôle pour lequel ce sera illisible ☐

Étudiant en école d'ingénieur en cybersécurité du logiciel, je recherche un stage en tant qu'apprenti ingénieur dans mon domaine d'étude pour perfectionner mes connaissances et en apprendre de nouvelles. À l'avenir je souhaiterais travailler dans l'univers de la cybersécurité même si je n'ai pas de projet précis pour le moment.

Études

2021-2024 ENSIBS (Vannes 56000) : École d'Ingénieur en cybersécurité du logiciel
2019-2021 Lycée Pothier (Orléans 45000) : CPGE PCSI/PCE (Physique-Chimie-Sciences de l'ingénieur / Physique-Chimie-Étoilé)

DAWIZZ

Figure 4 – Page obfuscation

3 Spécifications techniques

3.1 Choix technologiques

Besoins	Contraintes	Solution
Avoir accès au site	Rapide et sécurisé	FastAPI
Chiffrement/déchiffrement du texte	Rapidité d'exécution	Python
Algorithme chiffrement	Algorithme résistant au bruteforce	AES 128 GCM
Plateforme web	Facilement déployable	Docker

3.2 Accessibilité

3.2.1 Compatibilité navigateurs

Le site sera compatible avec les navigateurs suivants :

- ↗ Mozilla Firefox
- ↗ Google Chrome
- ↗ Safari
- ↗ Opéra

3.2.2 Types d'appareils

Le site sera conçu de manière dite “responsive” pour qu’il assure une navigation optimale sur tous types d’appareils :

- ↗ Téléphones mobiles
- ↗ Tablettes
- ↗ Ordinateur portables
- ↗ Ordinateur de bureau

3.3 Sécurité

- ↗ Mise en place de filtre pour empêcher l’envoi de fichier pdf/png vérolé.
- ↗ Utilisation d’un algorithme de chiffrement résistant.