# Compte Rendu Final

## Outil Automatique De Décryptage

Ce projet a été réalisé

par

Abdelaziz Ameurlain, Redha Dali, Mamadou Mouctar Barry, Allaye Afo Diallo, Rachid El Badaoui, Sofiane Hamad, Raphael Marouani, Linda Bedjaoui.

**Chef du projet fonctionnel :**
Linda Bedjaoui.

**Sous la direction de :**
**Madame Leila Kloul**

UVSQ
université PARIS-SACLAY

# Table des matières

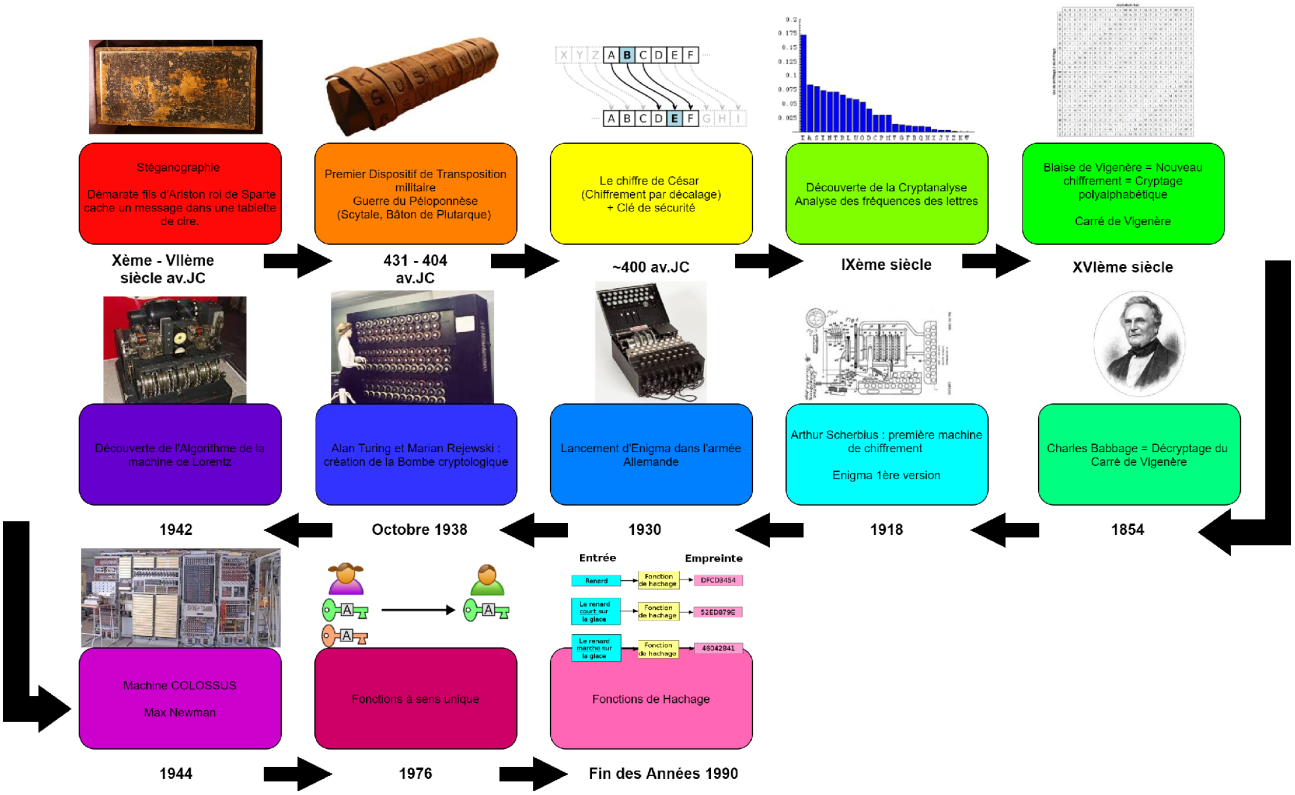
# 1 Introduction

L'application ODD (Outil automatique De Décryptage) vise à être accessible à n'importe quel utilisateur, qui voudrait chiffrer, déchiffrer ou décrypter des messages confidentiels. Deux choix s'offrent à elle pour l'utilisation de l'application : faire appel aux méthodes de Vigenère ou de Substitution. Toutes les fonctionnalités prévues lors de la phase de conception de l'application sont précisées dans ce document en indiquant l'implémentation de ces fonctionnalités dans l'application qui est programmée en langage C.

## 1.1 Historique

La Cryptographie est connue depuis des centaines d'années, mais reste encore étrangère au grand public. Avant de parler de Cryptographie, elle est intimement liée avec la Stéganographie, qui représente l'art de dissimuler un message. Débutant à l'antiquité, la Stéganographie puis la Cryptographie vont connaître un développement important. Peu après César utilisera un chiffrement par substitution qui lui sera propre, Il s'agit d'un chiffrement par décalage.

Ils surpassent la sécurité du chiffrement par décalage. Au XVIème siècle, Blaise de Vigenère, diplomate français propose un nouveau chiffrement soi-disant incassable. Arthur Scherbius, un ingénieur allemand invente alors une machine de chiffrement . Des échanges d'attaques et de défenses vont donner naissance à plusieurs machines qui joueront un rôle important dans le développement de la cryptographie. La découverte de nouvelles fonctions de chiffrement viendront complexifier les algorithmes déjà utilisés. La Cryptographie est un domaine très vaste qui s'est vu être enrichi depuis des millénaires, l'industrialisation des machines de cryptage a été le tournant dans la démocratisation des techniques de chiffrement.




## 1.2 Objectif de l'application

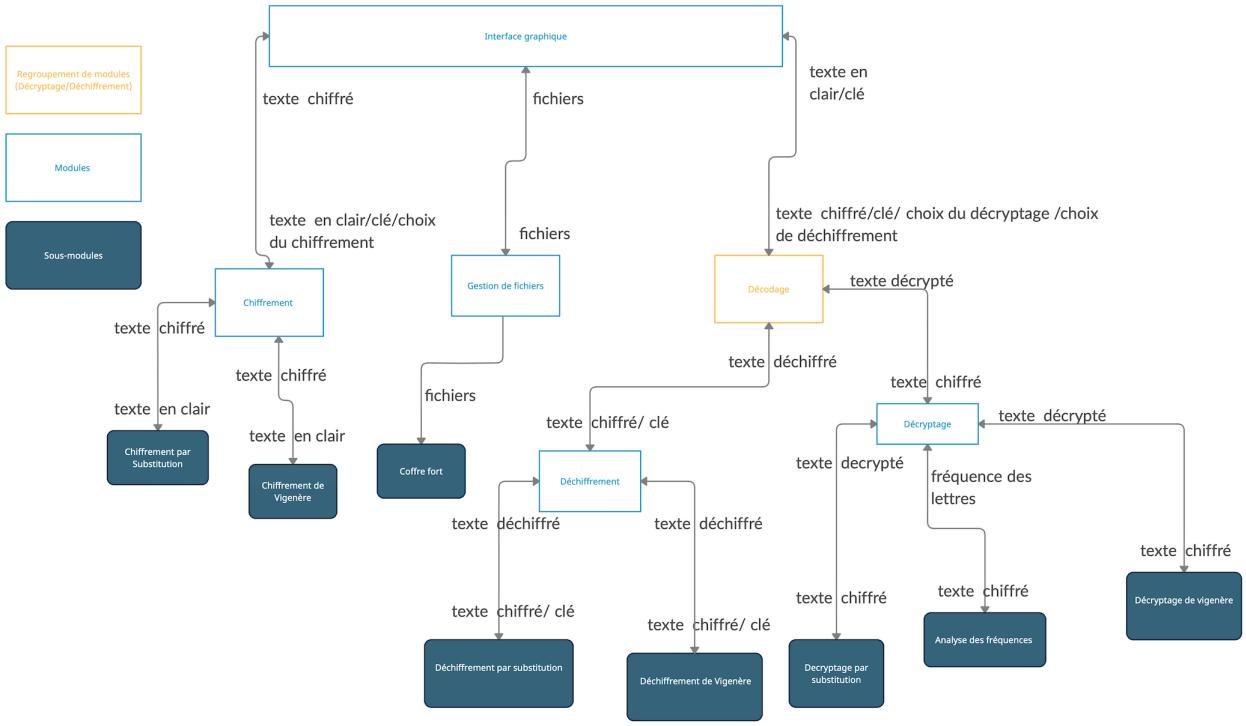
L'application doit être capable de crypter/déchiffrer du texte en utilisant le système Vigenère et / ou un système de cryptage par substitution. Elle peut également permettre la cryptanalyse (décryptage)

du texte crypté, c'est-à-dire finir par déchiffrer des messages chiffrés sans connaître au préalable la clé de cryptage. Elle permet aussi à l'utilisateur de faire une analyse de fréquences sur un texte quelconque (lettres ou paires de lettres) indépendamment.

**Apprendre le travail d'équipe**

Le projet se compose d'une équipe de 8 personnes, leur objectif est donc de coordonner le travail selon un calendrier hebdomadaire et de bien comprendre ses tâches, afin que la mise en oeuvre soit efficace et le client satisfait.

# 2 Organigramme




## 2.1 Objectif des modules

**Chiffrement :**

Ce module englobe les deux chiffrements Vigenère et dit de substitution. Il prendra en entrée des messages en clair et les chiffrera selon l'outil choisi.

**Déchiffrement :**

Ce module englobe les deux chiffrements Vigenère et dit de substitution. Il prendra en entrée des messages chiffrés et les déchiffrera selon l'outil choisi.

**Décryptage :**

Ce module englobe les deux chiffrements Vigenère et dit substitution. Il prendra en entrée des messages cryptés et les décryptera selon l'outil choisi. Il comporte également l'analyse de frequences d'un message

quelconque en entrée (lettre par lettre ou par paire de lettres). L'analyse fréquentiel est aussi utilisée dans le cadre de la cryptanalyse et ce de manière opaque du point de vue de l'utilisateur.

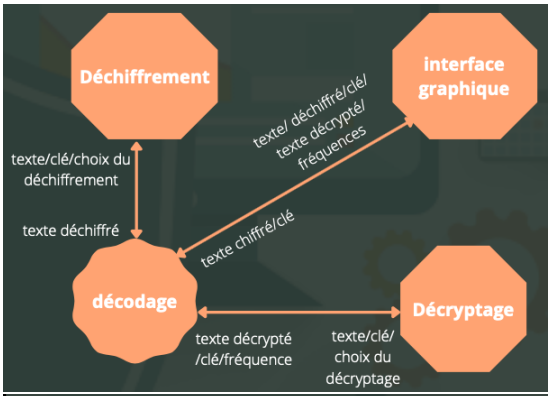
**Interface graphique :**

L'interface offre la possibilité d'écrire du texte en clair ou chiffré et de saisir une clé dans le cas d'un chiffrement ou d'un déchiffrement. L'interface sera dotée des boutons :

## 2.2 Explication des liens entre les modules

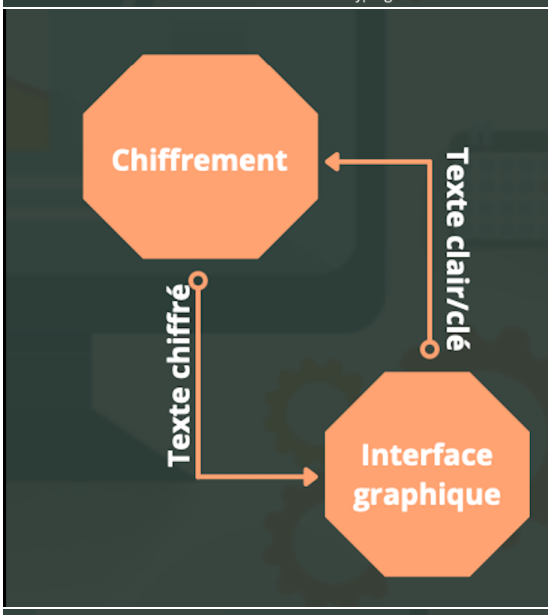
**Précision sur le regrouprement de modules (Décodage) :**

Décodage est un regroupement des modules Déchiffrement et Décryptage visant à rendre l'organigramme le plus clair possible visuellement. Il ne constitue pas un module à part entière.



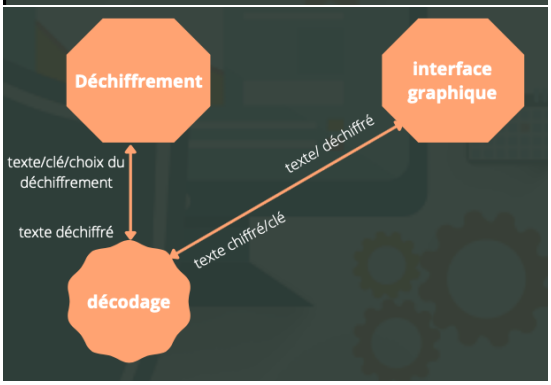

**Lien entre les modules Chiffrement et Interface graphique :**

L'interface graphique enverra le texte en clair ainsi que la clé de chiffrement en argument en appellant la fonction ChiffrementSubstitution ou ChiffrementVigenere selon le choix de l'utilisateur. L'interface graphique affichera ensuite à l'utilisateur le résultat du chiffrement retourné par la fonction appellée.



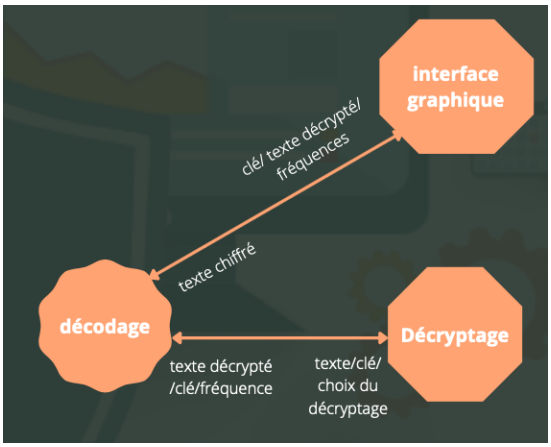

**Lien entre les modules Déchiffrement et Interface graphique :**

L'interface graphique enverra le texte chiffré ainsi que la clé de chiffrement en argument en appellant la fonction DechiffrementSubstitution ou DechiffrementVigenere selon le choix de l'utilisateur. L'interface graphique affichera ensuite à l'utilisateur le résultat du déchiffrement retourné par la fonction appellée.



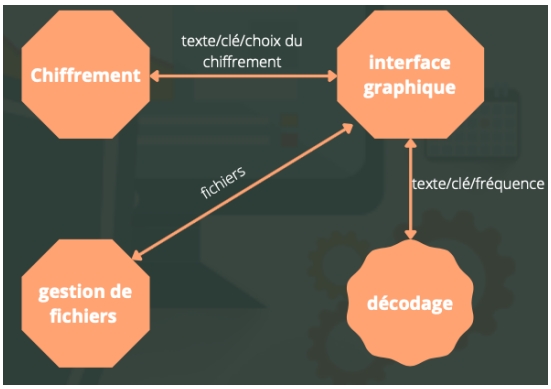
**Lien entre les modules Décryptage et Interface graphique :**

En cas de décryptage désiré par l'utilisateur et en fonction de son choix, l'interface graphique enverra le texte chiffré au module Décryptage en argument d'une des deux fonctions DecryptageSubstitution(texte chiffré) et DecryptageVigenere(texte chiffré) L'interface graphique pourra également envoyer un texte quelconque en argument de la fonction freqlettre* analyseFrequence(texte) si l'utilisateur souhaite simplement effectuer une analyse de fréquence. L'interface graphique affiche ensuite à l'utilisateur le résultat du décryptage ou de l'analyse de fréquence retourné par la fonction appellée.




**Lien gestion de fichiers et le module Interface graphique :**

L'interface graphique enverra uniquement le texte ou le texte et la clé au module sollicité en argument de la fonction représentant la fonctionnalité choisie par l'utilisateur. L'interface graphique affiche ensuite à l'utilisateur le résultat retourné par la fonction appellée et lui permettra de sauvegarder ou non le fichier obtenu.




# 3 Explication du choix du langage

Notre application sera basée sur la programmation procédurale, le cryptage ne peut donc pas être considéré comme une instance. C est un langage de bas niveau, ce qui signifie qu'il peut effectuer des opérations très proches du langage machine, ce qui peut accélérer le programme. De plus, c'est un langage qui peut fonctionner sur toutes les plateformes (Linux, MacOS, Windows), ce qui est bénéfique pour nos applications. Nous avons également choisi une bibliothèque graphique adaptée au langage C (GTK +). Cela nous permet de gérer efficacement la navigation entre les différentes fenêtres, ainsi que l'accès à la mémoire pour l'enregistrement et le chargement de fichiers.

# 4 Explication du fonctionnement de l'application ODD

L'application possède plusieurs caractéristiques :
- Chiffrement (par Substitution et Vigenère). - Déchiffrement (par Substitution et Vigenère). - Décryptage (par Substitution et Vigenère). - Analyse des Fréquences. - Coffre fort (sauvegarde de données).

L'utilisateur possède plusieurs options pour son utilisation, outre les différents outils à sa disposition pour chiffrer/déchiffrer/décrypter/faire l'analyse des fréquences de ses textes, il pourra également sauvegarder ses activités grâce à la fonctionnalité "Coffre-fort" accessible via l'interface.

**Chiffrement :**

Pour chiffrer un message, l'utilisateur va dans un premier temps sélectionner le bouton "chiffrer" avant de choisir l'outil de chiffrement voulu (Vigenère ou par substitution). Une fenêtre s'ouvrira lui permettant de saisir son message à chiffrer. Il lui suffira ensuite de cliquer sur le bouton "chiffrer" pour voir son message chiffré s'afficher dans une nouvelle fenêtre. Il pourra l'enregistrer s'il le souhaite ou quitter la fenêtre.

**Déchiffrement :**

Pour déchiffrer un message, l'utilisateur va dans un premier temps sélectionner le bouton "déchiffrer" avant de choisir l'outil de déchiffrement voulu (Vigenère ou par substitution). Une fenêtre s'ouvrira lui permettant de saisir son message à déchiffrer. Il lui suffira ensuite de cliquer sur le bouton "déchiffrer" pour voir son message déchiffré s'afficher dans une nouvelle fenêtre. Il pourra l'enregistrer s'il le souhaite ou quitter la fenêtre.

**Décryptage :**

Pour décrypter un message, l'utilisateur va dans un premier temps sélectionner le bouton "Décrypter" avant de choisir l'outil de cryptanalyse voulu (Vigenère ou par substitution). Une fenêtre s'ouvrira lui permettant de saisir son message à décrypter. Il lui suffira ensuite de cliquer sur le bouton "décrypter" pour voir son message en clair s'afficher dans une nouvelle fenêtre. Il pourra l'enregistrer s'il le souhaite ou quitter la fenêtre.

**Analyse des fréquences d'un texte :**

Pour faire l'analyse de fréquence d'un texte lettre par lettre ou par paire de lettre sur un texte en entrée, l'utilisateur va dans un premier temps sélectionner le bouton "Analyse fréquence" avant de choisir l'analyse voulu (lettre par lettre ou par paire de lettre). Une fenêtre s'ouvrira, lui permettant de saisir son texte. Il lui suffira ensuite de cliquer sur le bouton "analyser" pour voir son analyse s'afficher dans une nouvelle fenêtre. Il pourra l'enregistrer s'il le souhaite ou quitter la fenêtre. Lutilisateur peut effectuer cette analyse à n'importe quel moment, notamment pour des messages déjà chiffrés, déchiffrés ou décryptés par l'application. Un bouton "analyse" est disponible dans chaque fenêtre.

# 5 Partie technique

## 5.1 Point délicat :

Fonction **longueurCleIc** : Dans cette fonction calculant une taille de clé probable grâce aux indices de coïncidences à partir d'un texte chiffré par la méthode de Vigenère, il était initalement prévu dans notre pseudo-algo que nous allions effectuer les calculs d'indice de coïncidence pour chaque taille de clé possible jusqu'à atteindre la taille du message. Cependant, cela nous conduisait à des résultats d'indices de coïncidence à chaque fois peu cohérents et proche de 1. Cela faussait donc la longueur de clé trouvée à l'arrivée grâce à l'indice maximum et ne passait pas l'étape des tests unitaires. Nous avons par conséquent fixé une taille limite de clé au delà de laquelle les résultats pourraient commencer à devenir incohérents. Cette taille limite a été fixée à 7. Ce constat s'ajoute également au fait que les clés de chiffrement de Vigenère ne dépassent que rarement cette taille.

struct **freqlettre** : Nous avons ajouté un champ "int size" qui n'était pas présent dans le cahier des spécifications. Ce dernier nous permettra de récupérer la taille du tableau de struct (freqlettre), cela nous sera utile pour avoir le nombre de couples de lettres mais également pour mieux articuler la gestion de mémoire.

Fonction **analyseFrequenceCouple** : Cette fonction n'apparait ni dans le cahier des charges ni dans le cahier de spécification. Suite à une incompréhension dans la lecture du sujet, nous nous étions limité à l'implémentation d'une seule fonction d'analyse de fréquence lettre par lettre. Nous avons donc ajouté

cette fonction qui, comme spécifié dans le sujet du projet, effectue une analyse de fréquence par pair de lettres.

Fonction **rechercherCle** : cette fonction fonctionne de manière optimale uniquement lorsque la lettre la plus fréquente de chaque sous-chaîne du message découpé est la lettre E. Dans le cas contraire, il se peut que la clé calculée par la fonction comporte une ou plusieurs lettres incorrectes, ce qui peut fausser le décryptage. Il a été également nécessaire de définir une taille maximale de message et de clé en tant que variable globale, afin d'éviter tout problème d'allocation et à terme toute erreur de segmentation une fois que toutes les fonctions de décryptage sont appellées à la suite. La taille maximale de clé est fixé à 7 en adéquation avec la limite fixée par la fonction longueurCleIc. De plus, contrairement à ce qui était initialement prévu, la fonction rechercheCle n'utilise pas la fonction freqlettre* analyseFrequence. Cette dernière complexifiant significativement l'implémentation de la fonction sans pour autant apporter une plus value étant donné qu'ici, contrairement au décryptage par substitution, il n'est pas question de substituer par fréquence le texte entier mais simplement chaque lettre de la clé.

Fonction **decryptageSubstitution** : Cette fonction utilise une méthode de cryptanalyse afin de casser un chiffrement par substitution à l'aide d'une analyse de fréquences lettre par lettre et/ou par pair de lettres. La fréquence moyenne prise en compte de chaque lettre ou de chaque paire de lettre dans la langue française étant basée sur des milliers de texte, elle ne correspond que rarement à la réalité sur un texte donné. Pour remédier à ce problème, le programme nécessiterait une intelligence artificielle basée sur des algorithmes très avancées et capable de faire des choix lorsque la fréquence d'une lettre ou d'une paire de lettre dans le texte ne correspond pas à sa fréquence moyenne dans la langue française, cela afin d'aboutir à un texte correctement décrypté et intelligible en français. Par conséquent, la méthode de décryptage par analyse de fréquences du premier coup est très limitée. Afin de pallier ce problème, ou du moins d'obtenir le résultat le plus satisfaisant possible à l'aide de cette méthode, nous avons pensé à implémenter une troisième fonction d'analyse fréquentielle basée cette fois non pas sur des paires de lettres mais sur des groupes de trois lettres. Cela améliore le résultat mais ne résout pas pour autant entièrement le problème.

# 6 Organisation interne

La clé du succès dans un projet est l'organisation. Nous avons tenu à investir du temps dans la réflexion préliminaire avant de commencer et d'exécuter le projet et nommé Linda Bedjaoui en tant que responsable fonctionnelle, qui nous a présenté le programme hebdomadaire et les tâches à accomplir. Il y a de nombreux paramètres à maîtriser. Pour organiser méthodiquement un projet. Nous avons adopté un style de gestion plus ou moins approprié et utiliser de bonnes pratiques qui sont les suivantes :

**Au quotidien : ergonomie, agilité et système d'échange partagé :**

Nous avons créé un groupe via l'application Discord, et nous échangeons souvent nos questions, progrès et plannings. Certains d'entre nous se sont réunis à la bibliothèque de l'université. Nous avons également créé des comptes en communs sur différents sites pour que chacun puisse modifier ou ajouter quand il peut son travail : Overleaf qui est utilisé pour créer et modifier des documents en Latex, Canva qui sert à réaliser des slides et les plannings et Appcreatly pour les organigrammes.

**Périodique : Organiser des réunions (hôte) pour gérer l'avancement des tâches :**

Nous faisons une réunion zoom au moins une fois par semaine pour évaluer nos tâches, une réunion avant et après le cours de TD pour discuter des principaux points abordés avec l'enseignante Mme Kloul.

# 7 Conclusion

Le projet vise à créer une application ODD (outil de décryptage automatique) pouvant répondre aux attentes du client (Madame Leila Kloul). Quiconque souhaite crypter, déchiffrer et décrypter des messages confidentiels peut utiliser ODD. Pour concevoir cette application, nous utilisons deux outils principaux : les méthodes Vigenère et Substitution. En outre, l'analyse de fréquence (lettres ou paires de lettres) du texte peut également être effectuée indépendamment.